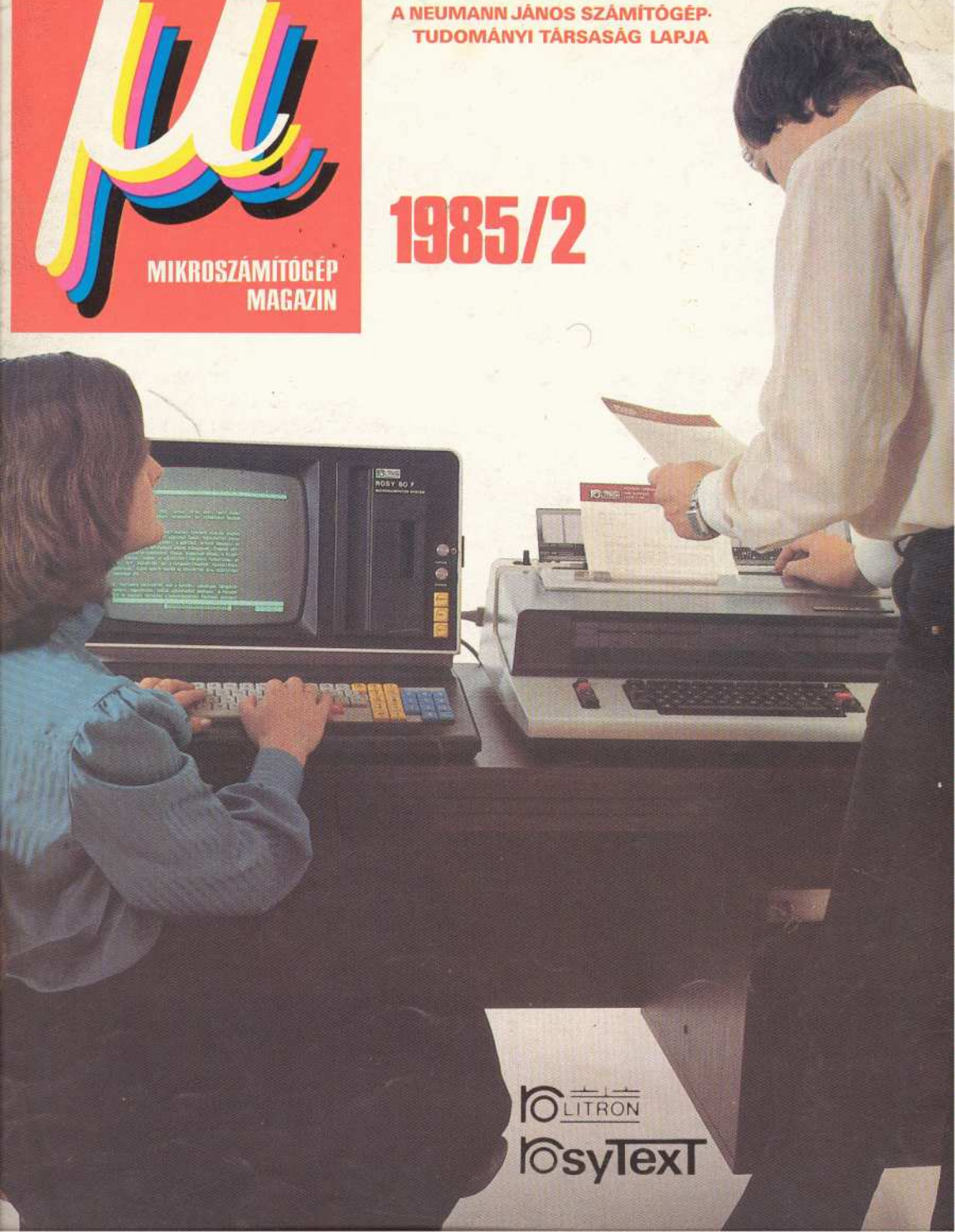
MIKROSZÁMÍTÓGÉP MAGAZIN

A NEUMANN JÁNOS SZÁMÍTÓGÉP-TUDOMÁNYI TÁRSASÁG LAPJA

1985/2



100 feladat – 1 programmal

Eddig csak 100 különböző programmal tudtak megoldani
Ma már mindezt

100 számítógépes feladatot a CSB–805 1 programmal oldja meg

a CSB–805 programrendszert számítástechnikai ismeretek nélkül is azonnal tudja használni

Commodore 64, VT–20/A, VT–20/IV, TZ–80, Transmic 8, Apple II és Multi Center számítógéprendszerein, hálózatain.

a CSB–805 segítségével Ön bármilyen nyilvántartási rendszerét elkészítheti, vagy a meglévő kartonrendszereit saját számítógépén megvalósíthatja

a CSB–805 nem igényel szakismeretet, speciális tanfolyamot, külön kiképzett számítógép-kezelőt, használatát bárki gyorsan elsajátíthatia

Tartsa nyilvántartásait CSB-vel naprakészen!

Fejlesztő: Automatika Ipari Kisszövetkezet
1133 Budapest, Gogol u. 22.
T.: 17-27-32

Részletes felvilágosítás a 260-534 telefonszámon

Forgalmazó:

SZ I SZ

ALTECH

Számítógépeit hálózatba szervezzük

A KALK–21 SEGÍTSÉGÉVEL PERCEK ALATT ÁTLÁTHATJA AZ ELKÉPZELT VÁLTOZÁSOK HATÁSAIT

Használata nem igényel számítástechnikai ismereteket

Minden gazdasági vezetőnek és munkatársnak segít döntései előkészítésében

MIKROSZÁMÍTÓGÉP MAGAZIN

A Neumann János Számítógéptudományi Társaság lapja

A kiadvány a Tudományszervezési és Informatikai Intézettel együttműködve készül

A szerkesztő bizottság vezetője:
Kovács Győző

Munkatársak:
Broczkó Péter (hírek)
Buday György István (személyi számítógépek)
Jakab Ágnes (ember–gép kapcsolat)
Kovács Győző (levelezés)
Lindner László (sakkprogramozás)
Pataki Ernő (programozástechnika)
Petróczy Judit (könyvek)
Pogány Csaba (alkalmazástechnika, tanfolyam)
Simonyi Endre (klub)
Takácsy Ildikó (favágás)
Varga András (iskola – számítógép)
Vass Nándor (alkalmazások)
Votisky Zsuzsa (játékprogramok)
Zárda Sarolta (piac)

A szerkesztőség tagjai:
Albert Tibor
Nacsa Sándor

Felelős szerkesztő:
Könyves Tóth Pál
Szerkesztőség:
1027 Budapest II., Fő utca 68.
Telefon: 154-250

Kiadja: a Delta Szaklapkiadó és Műszaki Szolgáltató Leányvállalat
Felelős kiadó:
Faklen Pál igazgató
1442 Budapest, VII., Garay utca 5.
Telefon: 415-583, 215-440

Terjeszti a Magyar Posta
Előfizethető bármely postahivatalban, a kézbesítőknél, a Posta hírlapüzleteiben és a Posta Központi Hírlap Irodában (Budapest V., József nádor tér 1. Postacím: 1900 Budapest) közvetlenül vagy postautalványon, valamint átutalással a PKHI 215–96162 pénzforgalmi jelzőszámra
Előfizetési díj:
egy évre 180,– Ft,
fél évre 90,– Ft.

Szedte:
a Nyomdaipari Fényszedő Üzem (857236/09)

Nyomás:
Petőfi Nyomda, Kecskemét, Külső Szegedi út 6. (85/50062)
Telefon: 28777
Felelős vezető:
Ablaka István igazgató

INDEX: 25629
ISSN 0236-6088

**Címképünk:
a Rolitron
Műszaki–Fejlesztő
Kisszövetkezet
Rosytext rendszere**

Tartalom

Copyright	2
Változó lehetőség, változó szükségszerűség	3
Adok – veszek – cserélek	9
Személyi számítógépek	22
Szoftver és alkalmazás	24
µprogramok	34
ISKOLA – SZÁMÍTÓGÉP	
Iskolaszámítógép ma és holnap	4
HT-BASIC DATA	4
Karakterfüzér nyomtatása memóriából	6
Gyors rajzoló játék HT–1080Z számítógépen	6
EMBER–GÉP KAPCSOLAT	
Vissza az ékezetekhez!	7
TERMÉKISMERTETŐ	
Easy Script	8
A Proper–16 személyi számítógép	13
PROGRAMOZÁSTECHNIKA	
Backtrack algoritmus – legkisebb mélységű keresés	10
TANFOLYAM	
Alapozás IX.	16
PIAC	
Hazai mikropiac '84	20
Az év gépe (Az év kísérlete)	20
µKLUB	
Klubújdonságok	28
Építsünk számítógépet! VII.	29
FAVÁGÁS	
Segédváltozós előállítás	33
JÁTÉKPROGRAMOK	
Másolóprogram	37
Folyosó	37
Hanoi-torony	38
Egy utasítással	39
Autóverseny – újra	39
AZ OLVASÓ ÍRJA	40
SAKKPROGRAMOZÁS	
Lépésről lépésre	42
KÖNYVEK	45
HÍREK, ÉRDEKESSÉGEK	47

Copyright

„Csak légy törvényes: azt ki kérdi,
Igazságos vagy-é vajon?
A böles dolgát tudja s érti:
Más otthon, más a piaczon:
Sok ember lett gyors fordulóban
Rossz, jó, meg' rossz, a mint szorult...
Hisz a világ nem ad méltóan
Sem pellengért, sem koszorút."
(Tompa Mihály)

Az Új Magyar Lexikonban a következőt találtam: „Copyright (kopirájt) <ang>: a szerzői jognak valamely műre vonatkozó megállapítása; a 'minden jog fenntartva' magyar kifejezés megfelelője. Rendszerint a könyvcímlap hátoldalán helyezkedik el. Jele: ©."

Joggal kérdezheti az olvasó, hogy mit akarok ezzel a kifejezéssel, miért írok erről, hiszen a szerzői jogot majdnem mindenki ismeri. Ha ennyi időm van, akkor inkább társadalmilag fontos problémákkal foglalkozzam, például olyasmivel, ami szakmai gond vagy valami hasonló. Hát ez is az, azt hiszem.

Egyik rovatszerkesztőnk hívta fel a figyelmemet a µM valamelyik számában megjelent hirdetésre, ahol egy olvasónk programokat ajánl. Miután szerkesztőnk fiát is érdekelte a dolog, írt a megadott címre, és meg is kapta a katalógust, amelyben az egyszerűbb játék- és egyéb programokból (100 forintos egységár) 30 darabot, a sakkprogramokból (300 forintos egységár) 6 darabot, a szuper játékprogramokból (500 forintos egységár) 36 darabot, a kezelői, illetve nyelvprogramokból 32 darabot talált!

A katalógus ezt írja: „A felsorolt programok egy része rendelkezik valamilyen (angol, német) nyelven írt kezelési utasítással. Másik részük feltételezi az adott nyelv vagy technika ismeretét. A programok egy részéhez rendelkezem fénymásolt kezelési útmutatóval (az x-szel jelzettek). Fénymásolási lehetőségeim korlátozott volta miatt csak kivételes esetben tudom a másolathoz hozzájuttatni Önöket.

Az összes programot ha kell, saját, ha kell, az általam vásárolt kazettára másolom. Természetesen, ha én veszem a kazettát, annak ára plusz költség a programhoz.

Minden vásárolt programot – ha esetleg hibásan van felvéve – természetesen kijavítom.

Ha valaki egyszerre legalább 5 programot vesz, akkor további egy programot ingyen kap hozzá! (Természetesen ugyanabból az árkategóriából.)

A másoló kapacitás függvényében időbe telik a kész kazetta elküldése, ezért előre is szíves türelmüket kérem!"

Körülbelül ennyi a nyolcoldalas stenciles katalógus szöveges része. A katalógusban felsorolt programnevek ismerősnek tűntek, miután valamelyik külföldi utam alkalmából egyikkelmásikkal különböző számítógépboltok és áruházak kirakatban találkoztam.

Persze a katalógusban is le van írva: „A felsorolt programok nagy része gyári, de sok van, melyeket listázni lehet, így sok olyan fogás tanulható belőlük, melyek a programok készítése során segítséget jelenthetnek. Sok programnak demonstrációs üzemmódja is van, illetve instrukciókat is tartalmaz."

Felhívtam a megadott telefonszámot és elmondtam, hogy a katalógusban ajánlott üzlet jogi háttere érdekelne, szeretnék erről a µM-ban írni, tehát az, hogy ezeknek az eredeti Sinclair programoknak eladása után hogyan is fizeti a jogdíjat a szerzőknek vagy a programok tulajdonosának.

Nincs okom kételkedni beszélgetőpartnerem jó szándékában és tisztességében, de azt hiszem, hogy egy, az amatőrök körében egyre jobban terjedő félreértés áldozata. Azt hitte ugyanis, sőt szakemberektől ilyen véleményt is kapott, hogy a legálisan (ajándékozással) birtokába jutott szoftverrel szabadon rendelkezhet, azt másolhatja, forgalmazhatja egy valójában igen kedvezményes árért – egyes boltok ugyanezekért a programokért az említett árak sokszorosát kérik! Azt is tudja, hogy a birtokában levő szoftver érték, és azt Sinclair Spectrumon nagyon jól lehet alkalmazni, ezért ezeknek a programoknak a népgazdasági hasznosítása közös érdek (itt elsősorban a kezelői és nyelvprogramokra gondolok).

Elmondta, hogy a szétküldött katalógusok nyomán azt hitte, hogy nem fog tudni eleget tenni a rendeléseknek, ilyen olcsón eredeti szoftvert kapni ajándék. Ehelyett alig volt rendelése, a gyér érdeklődés egy része sem végződött üzlettel. Közületek, iskolák, felsőfokú tanintézetek is megkeresték. Az utóbbiaknak ingyen másolta a programokat.

Úgy érezte, hogy nem kerül összeütközésbe semmiféle törvényes rendelkezéssel, hiszen – mint mondta – ezt az üzletet sokan, például klubok is csinálják. Erről egyébként én is hallottam, de eddig nem tudtam egyetlen eladóval vagy vevővel sem találkozni, ezért vagyok kénytelen írásomban beszélgetőpartnerem esetét használni példaként. Célom nem az, hogy őt felelősségre vonjam, inkább, hogy mindenkivel megértessem az ilyen tevékenység veszélyességét, és egyben azt is, hogy az ilyen üzlet a szakmai, most már írott törvények előírásaiba ütközik.

A szoftvert is a szerzői jog, az a bizonyos © betű védi, mint a hanglemezt, az írásművet vagy a videokazettát. Ha megveszem lemezen például Beethoven hegedűversenyét David Ojsztrah előadásában, és azt egy kiváló készüléken átmásolom kazettára és a kazettát árulom, akkor vétséget követek el, mert más szellemi tulajdonát bocsátom áruba anélkül, hogy annak, aki azt létrehozta, egy fillért is fizetnék. Ez akkor is igaz, ha például az említett kazettából kisebb a jövedelmem, mint a hanglemezgyárnak egy-egy lemez után. Pontosan így van ez a szoftverrel is: bárkinek az alkotását akkor lehet sokszorosítani és árulni, ha azzal a bárkivel megegyezem és szerződést írok alá.

Így igaz ez akkor is, ha a szoftver nem hazai gyártású, hanem külföldi eredetű, csak akkor a helyzet sokkal bonyolultabb. Külföldi partner esetében ugyanis szerződést csak érvényes külkereskedelmi joggal rendelkező szervezet írhat alá, arról nem is beszélve, hogy a tulajdonos részesedését devizában kell fizetni.

Aki a jelenlegi külkereskedelmi játékszabályokat és az ország devizagazdálkodását ismeri, és a cikket idáig elolvasta, most egy nagyot legyint, mert azt senki sem gondolja komolyan, hogy bárki ma játékprogramok szerzői jogainak megváltására egy centet is áldoz.

Akkor mi legyen a például Bécsben vásárolt és a vámhatóság tudomásával behozott programokkal, hogyan lehet ezeket terjeszteni? Üzleti vállalkozás formájában – véleményem szerint – sehogy. A vásárolt kazettát el lehet adni, el lehet cserélni, de másolni és pénzért árulni nem lehet.

Legutóbb egy müncheni boltossal beszélgettem, aki – mint elmondta – igen szép pénzeket fizet minden eladott kazettás szoftver után a felvételek tulajdonosainak. Megkérdeztem, hogyan tudja megakadályozni, hogy az általa eladott kazettát ne másolja le egy egész iskolára való gyerek. Azt mondta, hogy sehogyan, de nem is akarja. Szerinte nem jelent konkurrenciát, és főleg megakadályozhatatlan az ilyen cserebere. Az eredeti kazetta minősége, az adott garancia, bonyolultabb programok esetében a tanácsadás, betanítás és üzembe helyezés, mindez persze pénzért, a komoly vevőt mindig behozza az üzletbe. Sőt a sokak által üzött programcserebere még népszerűsíti is a terméket.

Azt hiszem, hogy a klubokban folyó kazettacsere nem üldözendő cselekmény, azt a számítástechnikai kultúra terjesztésének és nem az üzleti élet részének kell tekinteni. De fel kell lépnünk egységesen és nem elsősorban a törvény szigorával (az másnak a dolga), hanem etikai alapokon a bel- és külföldi szellemi termékek engedély nélküli árusítása ellen; szakmánk becsülete kívánja így.

Meg kell mondanom, hogy én is élvezettel nézem a Báthori utcai μklubban, amikor a legfrissebb μM-ból szigorú munkamegosztás szerint másolják a gyerekek a programokat. Egy óra múlva már mindegyiküknél kazettán van valamennyi program. A szerzőknek a μM fizeti a honoráriumot, megvásárolva ezzel a program használati jogát, amelyet ingyen ad át az olvasónak.

Sokan tudják, hogy számos kis alkotóközösség van az országban, azt hiszem, a legtöbb számítástechnikai területen dolgozik. Ezek nagy része programokkal is kereskedik. A gondos vevő nem teszi rosszul, ha ellenőrzi, joga van-e az eladónak, hogy az adott szoftvert árusítsa. Törvénytelen akciók esetén ugyanis a vevő nagyon rosszul járhat, hiszen egyetlen programhibával vagy hibás adathordozóval sem tud senkihez fordulni. Az eladó sokszor csak kezelni tudja a programot, a „lelkét" nem ismeri.

Végeredményben az az igazi védelme a szellemi terméknek, hogy azokat az esetek többségében csak a valódi tulajdonos tudja karbantartani, követni, újabb programrészekkel kiegészíteni vagy módosítani. A jogtalan forgalmazót főleg az üzleti haszon érdekli (most már régen nem a mi esetünkről beszélek), míg a gyártót vagy annak hivatalos képviselőjét a márka védelme is.

A nem hivatalos árusítás nemcsak az árbevétel elmaradása miatt veszélyes a gyártóra, hanem a márka megszerzett hírét, a „becsületét" is rontja. Ha a jogosulatlan eladó hibás programot ad el, esetleg nem tudja az eladott program hibáját kijavítani, a vevő azt mondja, hogy nem vesz több programot ilyen meg ilyen cégtől, és nem azt, hogy X. Y. nem ért a programok másolásához. A márkanév lejáratásának súlyos jogi és anyagi következményei lehetnek. Akit ez érdekel, legjobb, ha jogászhoz fordul, biztosan részletes és bőséges felvilágosítást fog kapni.

Elmélkedésemet a copyrightról ezzel be is fejezem. Még egyszer: nem akartam senkit sem pellengérre állítani, annál inkább egy kezdődő folyamatot vagy még inkább egy jelenséget. Egyre több gép van az országban, egyre több programra van szükség. Közös érdekünk, hogy a jövedelmek elosztását az eredmény, a teljesítmény határozza meg, és ne az ügyeskedés. Meg kell tanulnunk a tisztességes szoftverkereskedelmet, a törvények betartása valamennyiünk érdeke. A vevőké és az eladóké is.

Nem mondtam még el, mi volt telefonbeszélgetésünk vége. Partnerem a véleményemet kérte. Azt tanácsoltam, zárja be az üzletet, adja el vagy cserélje el a programokat, úgyis nagyobb gépre akar átváltani. Happy end.

KOVÁCS GYŐZŐ

Változó lehetőség, változó szükségszerűség

Úgy öt évvel ezelőtt, a technikai, gazdasági és alkalmazási összefüggések elemzéséből nyilvánvalóvá vált, hogy méreteit és intenzitását tekintve páratlan mikroszámítógépes forradalom előtt állunk. A most lezáruló, első szakaszban a személyi számítógépes technika térhódítása határozta meg a fejlődést. A gyártók szinte csak az egyén igényeit vették figyelembe, és csak a legalapvetőbb, hardver/operációs rendszer jellemzőket rögzítették.

A már megkezdődött, második szakaszban a rendszerbe integrálásra és a rendszerszintű funkcionalitás biztosítására helyeződik át a hangsúly. Sikeres gyártó csak az lehet, aki ebből kiindulva bővíti kínálatát, és ennek megfelelően tudja kézben tartani berendezései konstrukcióját. Végezetül, mindazon gyártók, akik nem képesek az alapberendezések milliós sorozatnagyságú előállítására (pl. személyi számítógép), visszaszorulnak a speciális kiegészítők, vagy az alapberendezéseiben a nagy gyártóktól vett, speciális rendszerek gyártásának területére.

Két oka van mindennek. A nyílt alapkonstrukció és a független fejlesztő, gyártó és értékesítő szervezetek közreműködésének ösztönzése eddig a legkedvezőbb volt a vezető szerepre törő gyártók számára. Elvezetett viszont egy nyitott piachoz és az ebből következő szerteágazó kínálathoz (közel 150 PC gyártó csak az USA-ban). A szoftver piacon kaotikus állapotok alakultak ki (10 ezer szoftver cég az USA-ban, összesen 32 ezer termékkel). Mindez elbizonytalanítja a vevőket, és mérsékli az eladások növekedési ütemét. A másik ok a nagy teljesítményű, új áramköri generáció (256 kbites RAM, 6–12 MHz-es és 16/32 bites mikroprocesszor), és a nagy kapacitású, új perifériák (Winchester lemez, optikai lemez, lézernyomtató) tömeges megjelenése, ami halaszthatatlanná teszi a rendszerszinten egységes megoldások bevezetését.

A korábbi stratégiát következetesen alkalmazó IBM elsőként kezdte meg az áttérést. Háromszoros processzor teljesítményű PC AT gépe, a 2 Mbit/s-os PC Network hálózat, a teljesen saját fejlesztésű, nagygépes adatkezeléssel integrált alkalmazási szoftverek és a TopView integrált alkalmazási környezet még csak előhírnökei a rendszerszinten átgondolt új termékek sokaságának. A hamarosan bevezetendő új PC generáció alacsonyabb árfekvése miatt a régi gépek árát jelentősen csökkentette a cég. Egy 256 kbájt RAM és 10 Mbájt Winchester tárkapacitású PC XT tényleges kiskereskedelmi ára 3000 font alá esett az angol piacon. Egy 512 kbájt RAM és 20 Mbájt Winchester tárkapacitású PC AT ára sem több 4700 fontnál.

Az Apple komplett irodai rendszerbe integrálja be Macintosh gépét. Az igen olcsó AppleTalk hálózatban (230,4 kbit/s) igénybe vehető LaserWrite (6995 $), nagy teljesítményű és nagy grafikai felbontású dokumentumnyomtatást tesz lehetővé. Az év közepén adatbázis kezelővel bővül a hálózati rendszer. A már eleve integrált alkalmazási környezetet nyújtó Macintosh-hoz „ellenőrzött" szoftvergyártók adják az egymással adatkommunikációra képes termékeket.

A legújabb Commodore gépek a szabványos megoldások bevezetésével közelítenek a rendszerbe illeszthetőség felé. A C128-as modellnek 410 kbájt floppy kapacitású CP/M üzemmódja is van a saját és a C64 üzemmód mellett. 80 oszlopos megjelenítés és 640 × 200 pontos, 16 színes grafika gondoskodik a teljesen professzionális kialakításról. A monitor nélküli konfiguráció irányára 1800, ill. 1000 nyugatnémet márka (floppyval vagy anélkül). A C200-as gép az IBM PC-vel kompatibilis. Irányára 5000 nyugatnémet márka.

A japán gyártók egy csoportja az MSX szoftver szabvány körül akarja kialakítani a házi számítógépek rendszerét. Az év közepén megjelenő második sorozat, a CP/M-mel kompatíbilis MSX–DOS változatban teljesen professzionális, és beintegrálódik a monitorként is használható, ún. digitális TV-be, valamint a képmagnó, képlemez és videotext rendszerekbe.

Ezzel szemben, a hazai fejlesztésű személyi számítógépek gyártása (a helyzetet érdemben nem befolyásoló PRIMO kivételével) még öt év alatt sem tudott eljutni a minimális, néhány ezer darabos sorozatnagyságig. A bizonytalan és kisvolumenű alkatrészellátás, a fontos perifériák hiánya és nem megfelelő minősége, még a legkomolyabb erőfeszítéseket is semmissé tette. Az így kialakult árszínvonal 5–10-szerese a megfelelő nyugatiénak. Mindez pedig jelentősen korlátozza a keresletet, egyre nehezebbé teszi az elért néhány száz darabos gyártás növelését, sőt még annak megtartását is veszélyezteti. Egyszóval, a hazai gyártásban egy tartós patthelyzet alakult ki.

Megfelelő szocialista kínálat sem volt, a tőkés behozatalt pedig az embargó és a drasztikus importkorlátozás akadályozta. A hazai ellátásnak így még a nyilvánvalóan szükséges igényekre sem telt erejéből. Például több, mint tízezer elektromechanikus könyvelőgép tölt be a normális gazdasági működéshez elengedhetetlen könyvelési, számlázási és más funkciót. A gépek teljesen elhasználódtak, egyre sürgetőbb felváltásuk egyetlen ésszerű formája a megfelelő mikroszámítógéppel való helyettesítés. Ehelyett maradt a kényszerhelyettesítés (pl. C64), illetve a mindenáron való életben tartás bizonytalansága.

A szerény hazai eredmények, de különösen a most kialakuló új körülmények miatt, a hazai gyártást és ellátást teljesen új alapokra kellene helyezni. A hazai gyártóbázis strukturális fejlődését is csak az biztosíthatja, ha nem a kommersz alapberendezések kis sorozatú előállítására összpontosítja erőit. A C128 vagy MSX–DOS kategóriájú gépek sem embargósak többé. A probléma felismerése mellett minden érintettnek alaposan elemeznie kell a helyzetet. Ehhez kíván hozzájárulni a μM mostani száma két elemzés közzétételével. Az elsőben a személyi számítógépek piacon kialakult kategóriáiról, gyártási és értékesítési viszonyairól adunk általános tájékoztatást. A másodikban a szoftver ellátást és az ezzel szorosan összefüggő alkalmazási kérdéseket vizsgáljuk.

NACSA SÁNDOR

Iskolaszámítógép ma és holnap

Már több, mint három éve, hogy az első iskolaszámítógépek, mint az új technika és a folyamatosan megújuló oktatás eszközei, megjelentek a középiskolákban. A szocialista országokban elsőként – a világ élvonalához képest sem túl későn – sikerült bevezetnünk és általánosan elterjesztenünk a középiskolai számítástechnikai oktatást.

Ma már az ország minden középiskolájában legalább egy, de inkább több gép áll a tanulók és oktatók rendelkezésére, hogy minden tanuló elsajátítsa a XX. század új tudományágának alapjait, ami hamarosan az általános műveltség részévé válik.

A gép – a HT–1080Z iskolaszámítógép – és a gyártó – a Híradástechnika Szövetkezet – körül kialakult viták, nézeteltérések régen elcsitultak, a nemes cél és az elért eredmények alaptalanná tették a tamáskodók fenntartásait. A tanulók és tanárok elfogadták a gépet, és nagy lelkesedéssel használják munkájukban.

A Híradástechnika Szövetkezet elkötelezte magát, hogy részt vállal az oktatásügy eme nagy feladatában, és az igényeknek megfelelően gyártja és fejleszti tovább az iskolaszámítógépet.

Első lépésként az alapgépet fejlesztette tovább: bevezette az ékezetes karaktereket és 64 kbájtra bővítette a beépített memóriát.

Ezután a perifériális egységek következtek. Ezek sorában az első és legszükségesebb eszköz a nyomtató, amellyel a programlisták és számítási eredmények rögzíthetők.

A *nyomtatóillesztő egység* (HT–1080Z/PR1) kifejlesztésével lehetővé vált az ún. Centronix rendszerű nyomtatóknak a géphez csatlakoztatása. Ez az illesztőegység önmagában használható, tápfeszültség-ellátást az alapgépből kap, beépített csatlakozó kábeleivel kapcsolható a számítógéphez és a nyomtatóhoz. Lehetővé teszi egy további egységnek a számítógép buszrendszeréhez csatlakoztatását is.

Az illesztőegység jelenleg is beszerezhető. Az önállóan csatlakoztatható illesztőegységek számát korlátozza a számítógép beépített tápegységének kapacitása. Ezért ha többféle illesztőegység csatlakoztatására kell felkészülni, akkor feltétlenül szükség van az újonnan kifejlesztett bővítő egységre.

A *bővítő egység* (HT–1080Z/EU7) alkalmazásával 7 db egykártyás kiegészítő vagy interfész kártya csatlakoztatható. Biztosítja a szükséges tápellátást, a buszkapcsolatot, és megteremti a rendszer moduláris bővítésének lehetőségét. Ezt az egységet az iskolák 1985. IV. negyedévében vásárolhatják meg.

A bővítő egységbe tolható, eddig kifejlesztett vagy fejlesztés alatt álló kártyák az alábbiak.

A *nyomtatóillesztő* (HT–1080Z/PR2) szolgáltatásaiban megegyezik az önállóan használható nyomtatóillesztővel, de értelemszerűen további buszcsatlakozást nem biztosít, hiszen erre a bővítő egység szolgál.

A *finom grafika egység* (HT–1080Z/HRG) alkalmazásával az alapgép korlátozott grafikai szolgáltatásait lehet nagymértékben fokozni. Vízszintes és függőleges irányban egyaránt 512 pontos felbontást biztosít, képpontonként kétbites gradációval. Az alapgéptől független, egy teljes kép tárolására szolgáló tárkapacitással egészíti ki a rendszert. Így széles variációs lehetőséget nyújt a teljes képmezőre, és egyben tehermentesíti az alapgép tárát.

Az egység közvetlenül BASIC programból vezérelhető, és sokféle grafikus manipulációra ad lehetőséget: vízszintes és függőleges irányú nagyítás, korlátlan origóáthelyezés, négyirányú „roll" funkció, képpont-visszaolvasás, invertálás, direkt törlés stb.

Felépítése olyan, hogy szerves részét képezi az 1986-ban forgalomba kerülő színes egységnek is, tehát egy kiegészítő kártya alkalmazásával lehetővé válik a rendszer színes grafikus felhasználása. Természetesen ezt a kiegészítő kártyát is a bővítő egységbe lehet majd betolni.

A *hajlékony mágneslemez-háttértároló-illesztő egység* (HT–1080Z/FI) lehetővé teszi a hajlékonylemezes háttértárolók alkalmazását a rendszerben. Az egységbe beépített, illetve kívülről hozzákapcsolható kazettás tároló kapacitása és felhasználási lehetősége is korlátozott (soros keresés, hozzáférési idő). Ahhoz, hogy az iskolaszámítógép felhasználási területét ki lehessen bővíteni, elengedhetetlenül szükséges egy nagyobb teljesítményű háttértároló, a hajlékonylemez üzembe állítása.

Az illesztőegység maximálisan 4 db lemezegység kezelésére alkalmas. Az illesztőegységgel, illetve a lemezegységekkel együtt a felhasználóknak egy igen aktívan és nagy hatásfokkal használható lemezrendszer alkalmazására nyílik lehetőségük.

A *hajlékony mágneslemez egység* (HT–1080Z/FD) önálló egységként, külön dobozban helyezkedik el. 5 1/4"-os normál írássűrűségű, egyoldalas adatfelírással 0,875 Mbit adatmennyiség tárolható egy lemezen, a hozzáférési idő kb. 600 ms, a maximális adatátviteli sebesség 125 kbit/s. A legfeljebb 4 lemezegység használata esetén az egyes egységek láncolva kapcsolhatók össze egymással, illetve az illesztőegységgel.

A felsorolt kártyák és a lemezegység a bővítő egységgel együtt ez év IV. negyedévében kerül forgalomba.

Szoftverbővítésként a FORTH programozási nyelv áll a felhasználók rendelkezésére. Felhasználásával kiterjeszthetők a gép szolgáltatásai mind az oktatási, mind a feladatmegoldások területén.

Az iskolaszámítógép fejlesztése természetesen nem áll meg. További bővítő és interfész egységek megjelenése várható (RS232 interfész, IEC interfész, PROM programozó és betöltő, színes egység, hálózatilánc-interfész stb.).

A gyártási program is folyamatos, hamarosan lehetőség nyílik arra, hogy egyéb felhasználók, nem oktatási célokra is vásárolhassanak megfelelő számban ebből a számítógépből.

Az iskolaszámítógép-program túljutott az első lépéseken. Az eredmények egyértelműen pozitívak, a számítástechnika bevonult a tanulók mindennapi életébe. A program kiszélesítéséért minden érdekeltnek teljes lendülettel tovább kell dolgoznia, hogy az eredményes indulást megfelelő folytatás kövesse.

ERDÉSZ ISTVÁN

HT–BASIC–DATA

A HT iskolaszámítógép BASIC-jének egyik hibája, hogy a DATA-k olvasását nem lehet egy tetszőleges számú sortól kezdeni, csak a legelejétől. Nézzünk egy példát. Van egy programunk, amelyben sok konstans adatot kell eltárolnunk. Ezt csak DATA-val tehetjük meg:

```
10 DATA 4,1,5,2,5,8,2,4,3,0,11,5,7,6,9
20 DATA 7,10,6,9,6,3,9,7,8,11,0,6,4,5,2
.
.
.
150 DATA 2,–1,3,0,3,6,0,2,1,–3,9,3,5,4,7
```

Hogyan oldhatjuk meg a 150-es sor elemeinek kiolvasását? Csak úgy, hogy előtte kiolvassuk a többi – számunkra fölösleges – 14 sor elemeit is, majd mikor az olvasás a 150-es sorhoz ért, akkor tároljuk a számunkra már értékes adatokat.

Programban:

```
1000 FOR A=1 TO 14*15 : REM Az első DATA elemtől a 14-es sor utolsó eleméig
1010 READ B           : REM Itt csak maga az olvasás lényeges, a kiolvasott adat nem
1020 NEXT A
1030 READ E           : REM Most kezdődik a számunkra érdekes elemek kiolvasása
```

Ha kipróbálnánk ezt a programot, azt tapasztalnánk, hogy nagyon lassan fut. Nem is csoda, hiszen egy 1-től 210-ig futó ciklus – benne egy READ-del – elég sok időt foglal le. Hogyan lehetne megoldani ezt a problémát?

Ehhez egy kis kitérőt kell tenni. Meg kell tudni, hogyan tárolja a számítógép a BASIC programot.

A gép a programot a 17129 (42E9 hexadecimálisan) címtől kezdve tárolja *(1. táblázat)*. Nézzünk egy példát! Hogyan tárolódik ez a rövid program?

```
10 REM >>>>
20 PRINT A
```

Ha jobban áttanulmányozzuk a táblázatot, akkor fel tudunk írni egy általános formátumot egy programsor tárolására.

1.–2. bájt – a következő sor kezdőcíme
3.–4. bájt – a programsor sorszáma
4.–n. bájt – utasítások, argumentumok
(n+1.) bájt – sor vége vagy program vége jel

Már a táblázatból is észrevehettük, hogy például az INPUT utasítást nem betűnként (I; N; P; U; T), hanem egyetlen szám formájában

Memóriacím		Ezen a címen tárolt érték		Megjegyzés	
Hexdec.-ben	Dec.-ben	Hexdec.-ben	Dec.-ben		
42E9	17129	F5	245	A következő programsor ettől	
42EA	17130	42	66	a címtől kezdve helyezkedik el	
42EB	17131	0A	10	A programsor sorszáma	
42EC	17132	00	0		
42ED	17133	93	147	A REM utasításkódja	
42EE	17134	20	32	A szóköz ASCII kódja	
42EF	17135	3C	60	A > jel ASCII kódja	
42F0	17136	3C	60	A > jel ASCII kódja	argumentum
42F1	17137	3C	60	A > jel ASCII kódja	
42F2	17138	3C	60	A > jel ASCII kódja	
42F3	17139	20	32	A szóköz ASCII kódja	
42F4	17136	00	0	Sor vége jel	
42F5	17141	FD	253	A következő programsor	
42F6	17142	42	66	kezdőcíme	
42F7	17143	14	20	A programsor sorszáma	
42F8	17144	00	0		
42F9	17145	B2	178	A PRINT utasításkódja	
42FA	17146	20	32	Szóköz	argumentum
42FB	17147	41	65	Az A karakter ASCII kódja	
42FC	17148	00	0	Sor vége jel	
42FD	17149	00	0	Program vége jel	
42FE	17150	00	0		

1. táblázat

128 END	171 LSET	214 <
129 FOR	172 RSET	215 SGN
130 RESET	173 SAVE	216 INT
131 SET	174 SYSTEM	217 ABS
132 CLS	175 LPRINT	218 FRE
133 CMD	176 DEF	219 INP
134 RANDOM	177 POKE	220 POS
135 NEXT	178 PRINT	221 SQR
136 DATA	179 CONT	222 RND
137 INPUT	180 LIST	223 LOG
138 DIM	181 LLIST	224 EXP
139 READ	182 DELETE	225 COS
140 LET	183 AUTO	226 SIN
141 GOTO	184 CLEAR	227 TAN
142 RUN	185 CLOAD	228 ATN
143 IF	186 CSAVE	229 PEEK
144 RESTORE	187 NEW	230 CVI
145 GOSUB	188 TAB (	231 CVS
146 RETURN	189 TO	232 CVD
147 REM	190 FN	233 EOF
148 STOP	191 USING	234 LOC
149 ELSE	192 VARPTR	235 LOF
150 TRON	193 USR	236 MKI$
151 TROFF	194 ERL	237 MKS$
152 DEFSTR	195 ERR	238 MKD$
153 DEFINT	196 STRING$	239 CINT
154 DEFSNG	197 INSTR	240 CSNG
155 DEFDBL	198 POINT	241 CDBL
156 LINE	199 TIME$	242 FIX
157 EDIT	200 MEM	243 LEN
158 ERROR	201 INKEY$	244 STR$
159 RESUME	202 THEN	245 VAL
160 OUT	203 NOT	246 ASC
161 ON	204 STEP	247 CHR$
162 OPEN	205 +	248 LEFT$
163 FIELD	206 –	249 RIGHT$
164 GET	207 *	250 MID$
165 PUT	208 /	255 ISA
166 CLOSE	209 [	
167 LOAD	210 AND	
168 MERGE	211 OR	
169 NAME	212 >	
170 KILL	213 =	

2. táblázat

tárolja. A *2. táblázatban* a számítógép teljes alaputasítás-készletét és a hozzá rendelt utasításkódokat láthatjuk.

Az utasításkódok 128-tól kezdődnek. Ez értheto, hiszen a programba argumentumként beírható karakterek ASCII kódja 127-ig terjed (grafikus karaktereket nem tudunk billentyűzetről közvetlenül a programba beírni, például REM után). Ha átfedés lenne az argumentumot jelentő kódok és az utasítást jelentő kódok között, akkor a gép nem tudná eldönteni, utasítással vagy argumentummal áll-e szemben.

És most kanyarodjunk vissza az eredeti kérdéshez, a DATA-k olvasásához. Mi lenne, ha az összes DATA utasításkódját átírnánk a REM utasításkódjára? Csak azt írnánk vissza a DATA kódjára, amelyikből olvasni akarunk. Ha abbahagytuk a kiszemelt DATA (REM) olvasását, akkor visszaírnánk REM-re. Ez bizony célravezetőbb módszer. Hiszen csak az a DATA valódi – élő – DATA sor, amelyikből olvasni akarunk. Így nem kell az a hosszadalmas elő-kiolvasás.

Így tulajdonképpen két „műveletet" kell elvégezni: REM átírását DATA-ra, s ha végeztünk, DATA átírását REM-re.

Ez legcélszerűbben két szubrutinnal oldhatjuk meg. A szubrutinoknak kezdő adatként az átírandó sor sorszámát kell megadni az 'A' változóban.

Az első rutin a REM-ről DATA-ra átírást végzi.

```
1000 B=17129
1010 C=PEEK(B)+PEEK(B+1)*256:D=
     PEEK(B+4)
1020 IF C=0 THEN PRINT
     „UL ERROR IN RESTORE SUBROUTINE": END
1030 IF PEEK (B+2)+PEEK(B+3)*256=A
     THEN IF D=147 OR 136 THEN POKE B+4, 136
     :RESTORE:RETURN ELSE PRINT
     „FC ERROR IN RESTORE SUBROUTINE"
     :END ELSE B=C:GOTO 1000
```

A második rutin a DATA-ról a REM-re írást végzi.

```
2000 B=17129
2010 C=PEEK (B)+PEEK(B+1)*256
     :D=PEEK(B+4)
2020 IF C=0 THEN PRINT
     „UL ERROR IN RESTORE SUBROUTINE": END
2030 IF PEEK (B+2)+PEEK (B+3)*256 = A THEN IF
     D=136 OR D=147 THEN POKE B+4, 147
     :RESTORE:RETURN ELSE PRINT
     „FC ERROR IN RESTORE SUBROUTINE":END
     ELSE B=C:GOTO 2010
```

A rutinokat természetesen GOSUB 1000, illetve GOSUB 2000 utasításokkal hívjuk meg.

Ha már meglévő programba akarjuk beépíteni a szubrutinokat, akkor más sorszámtól, más változókkal is be lehet írni, csak arra kell vigyázni, nehogy összekeverjük a változókat.

BESENYEI PÉTER

Karakterfüzér nyomtatása memóriából

```
        KARAKTERSOR NYOMTATASA MEMORIABOL

10 'MEGADOTT CIMTOL CIMIG ASCII-KODOT VISZ KI NYOMTATORA
20 '
30 ' LAKAT TAMAS 1984.NOV.7
40 '
50 CLEAR 500
60 INPRINT"KEZDOCIM";A
70 INPRINT"VEGCIM";Z
80 A$="                                                              "
90 Z=Z+1: K=0: B=VARPTR(A$)
100 IF K=0 THEN GOSUB 200
110 L=Z-A: IF L>254 THEN L=64
120 POKE B,L
130 A1=INT(A/256):A2=A-256*A1
140 POKE B+1,A2:POKE B+2,A1
150 LPRINT "(  " A$ "  )"
160 K=K+1: IF K=17 THEN GOSUB 220
170 A=A+L: IF A<Z THEN 100
180 IF K>0 THEN GOSUB 220
190 END
200 LPRINT:LPRINT STRING$(70,42):LPRINT"*" STRING$(68,32);"*"
210 K=1:RETURN
220 LPRINT"*";STRING$(68,32);"*":LPRINT STRING$(70,42):LPRINT
230 K=0:RETURN
```

FUTTATAS

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+                                                                      +
+  KEZDOCIM? 15360                                                     +
+  VEGCIM? 15400                                                       +
+  READY                                                               +
+  LIST                                                                +
+  READY                                                               +
+  LPRINTCHR$(14):LPRINT"    KARAKTERSOR NYOMTATASA MEMORIABOL "       +
+  READY                                                               +
+  LPRINTCHR$(15):LPRINT:LLIST                                         +
+  READY                                                               +
+  LPRINT:LPRINTCHR$(14):LPRINT"              FUTTATAS":LPRINTCHR$(15) +
+                                                                      +
+  READY                                                               +
+  RUN                                                                 +
+  KEZDOCIM? 15360                                                     +
+  VEGCIM? 15400                                                       +
+                                                                      +
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++


++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+                                                                      +
+  SSEGSOMMV  *                                                        +
+                                                                      +
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

A fenti program segítségével a RAM tetszőleges tartományából lehet ASCII kódokat nyomtatóra kivinni. Ez beépíthető rutinként szövegkezelő, rajzoló, szerkesztő programokba, ahol általában a szerkesztendő szöveget célszerű egy képernyő méretű (1 kbájt) tárolóban elhelyezni. A program úgy készült, hogy ezt a képernyőméretet vette számításba. A nyomtatóra 64 karakterenként viszi ki a szöveget, majd sort emel, 16 soronként pedig bekeretezi a szöveget. Ha sor közben fejezi be a nyomtatást, akkor két szóköz után csillaggal jelzi a végét.

A 80-as sorban definiált Q$ változó hosszát és kezdőcímét a 120-as és 140-es sorokban POKE segítségével módosítjuk. A hosszt az L változóban tároljuk, értékét a 110-es sorban a kezdő- és végcím különbségeként, de legfeljebb 64-nek állapítjuk meg. A 150-es sorban – két csillag között – nyomtatóra küldjük az LPRINT Q$ segítségével az aktuális 64 karaktert. A 110, 160 és ennek megfelelően a 200 és 220-as sorok megváltoztatásával más formátumot is beállíthatunk.

LÁNYI TAMÁS

Gyors rajzoló játék HT–1080Z számítógépen

A HT–1080Z számítógépen az IF utasítás nagyon lassú. Így ha azt az egyszerű rajzoló játékot, amely figyeli a billentyűzetet és négy irányban rajzol, töröl, IF utasítások felhasználásával írjuk meg, akkor keservesen lassan mozog a pont. Ha viszont INKEY$ helyett közvetlenül a billentyűzetmátrixból olvasunk és a beérkező kódokat IF helyett logikai műveletekkel vizsgáljuk, akkor ugrásszerűen megnő a rajzolás sebessége.

Az alábbi program a négy nyíl billentyű megnyomásával rajzol, a pont ferdén is mozoghat. LŐRENTEI TAMÁS

```
10 CLS : DEFINT A,X,Y
20 A=PEEK(14400)
30 X=X-(X < 63)*(A AND 64)/64+
   (X > 0)*(A AND 32)/32
40 Y=Y-(Y < 47)*(A AND 16)/16+
   (Y > 0)*(A AND 8)/8
50 RESET ((A AND 1)=0)*(-64)+X,Y)
60 SET (X,Y) : GOTO 20

READY.
```

```
10 CLS : DEFINT A,X,Y
20 A=PEEK(14400)
30 X=X-(X<63)*(A AND 64)/64+
   (X<0)*(A AND 32)/32
40 Y=Y-(Y<47)*(A AND 16)/16+
   (Y>0)*(A AND 8)/8
50 RESET ((A AND 1)=0)*(-64)+X,Y)
60 SET (X,Y) : GOTO 20

READY.
```

```
10 CLS : DEFINT A,X,Y
20 A=PEEK(14400)
30 X=X-(X<63)*(AAND64)/64+
   (X<0)*(AAND32)/32
40 Y=Y-(Y<47)*(AAND16)/16+
   (Y>0)*(AAND8)/8
50 RESET ((AAND1)=0)*(-64)+X,Y)
60 SET (X,Y) : GOTO 20

READY.
```

T. SZERZŐ KOLLÉGÁK!

A honoráriumok zökkenőmentes átutalása érdekében kérjük, hogy az írásokkal, cikkekkel együtt az alábbi adatokat is szerkesztőségünkbe elküldeni szíveskedjenek:

Név
Születési hely, idő
Anyja neve
Lakáscím, telefonszám
Munkahely megnevezése
Munkahelyi címe, telefonszáma
Személyi száma
Utalás esetén melyik címre kéri a honoráriumot

Vissza az ékezetekhez!

Két hozzászólást is kaptunk 1984. évi 4. számunkban megjelent cikkünkhöz. Eredeti formájukban tesszük közzé mindkettőt, mivel „ékezetesen" bizonyítják, hogy a szövegfeldolgozás funkciójában már van megoldás. Bárcsak más feldolgozási funkcióban is így állna a helyzet. Példaként említhetjük az adatbázissal támogatott gépi nyilvántartást. Az adatbázis lekérdezésénél, illetve tablók készítésénél az adatbázisból, egyáltalán nem mindegy, hogy milyen sorrend szerint rendezettek az egyes tételek. Az „Á" betűvel kezdődő kulcsszavú tételeknek például közvetlenül az „A" betűvel kezdődők után kell lenniük. Az eddig ismert adatbázis kezelők a kódtáblázat szerinti sorrendet veszik figyelembe, az pedig egészen más. A szoftverek többsége a „tok" és a „tök" között sem tud különbséget tenni, eléggé tökkelütöttek, ugye? Egyszóval, várjuk további hozzászólásaikat, javaslataikat és megoldásaikat.

Tisztelt Mikroszámitógép Magazin!

Érdeklődéssel olvastam a **Pont, pont, vesszőcske** című cikket 1984. évi 4. számukban. Miután ezzel a problémával magam is találkoztam, ezért utánanéztem és beszereztem néhány olyan információt, amelyek egy lehetséges megoldás körvonalait villantották fel előttem, és ezt most az önök figyelmébe ajánlom.

Ma a számítástechnikában az egyik legelterjedtebb kódolási rendszer az un. **"ASCII"** kódrendszer. Ebben a kódtáblázatban természetesen az amerikai gyakorlatnak megfelelő karaktereket találhatjuk meg, a magyar karaktereket nem. Ez a karakterválaszték többnyire megfelel a nemzetközi számítástechnikai gyakorlatban, de egy más betüket is tartalmazó ABC-t használó nemzet számára (pl.: levelek irásakor) szüknek bizonyul. Ezt az akadályt igyekszik általában mindenki a saját eszközeivel kiküszöbölni. Ennek megfelelően az egyes országokban *nemzeti karakterkészletet* határoznak meg számítástechnikai alkalmazásokra. Ez azt jelenti, hogy bár alapul veszik az emlitett ASCII karakterkészletet, **de** több helyen megváltoztatják azt az igényeknek megfelelően. Magyarországon az **MSZ 9212**-ben találhatjuk meg az ASCII kódkészlet magyar ékezetes betükkel kiegészitett szabványát.

Az igényesebb számítástechnikai berendezések gyártói szinte kivétel nélkül ellátják termékeiket nemzeti karakterkészletekkel. A magyar televizió képujságjában azért a svéd ABC betüit olvashatjuk, mert ez a nemzeti karakterkészlet áll a legközelebb a magyar betükészlethez azok közül, amelyeket egyáltalán alkalmaznak a gyártók.

Itt merül fel tehát az egyik lehetséges megoldás: olyan eszközöknél, ahol az megtalálható, használjuk a svéd ABC megfelelő betüit. Ez azonban távolról sem teljeskörü megoldás, arról nem is beszélve, hogy nem tesz eleget az MSZ 9212 előirásainak.

Aki ezt a szöveget olvassa, bizonyára észrevette (annak ellenére, hogy láthatóan számítástechnikai eszközökön készült), hogy a magyar ékezetes betük csaknem mindegyike szerepel benne, ugy, ahogy azt megszoktuk. Talán csak az okoz meglepetést, ha azt is elárulom, hogy a szöveg egy SINCLAIR Spectrum-on iródott. Irás közben a képernyőn ugyanugy látszottak az ékezetes betük, mint itt a papiron, és ha ugy kivánja a szöveg, akkor az ASCII szabvány szerinti összes betü **is** megjelenhet mind a képernyőn, mind a nyomtatott szövegben (pl.: []~¦\{} stb.). A másik meglepetést az jelentheti, hogy a kiirásnál használt nyomtató lehetett volna akár SEIKOSHA, akár EPSON vagy más gyártmányu mátrixnyomtató; ez a leirás konkrétan egy EPSON RX-80-as nyomtatón készült.

és mindezek a magyar szabványnak megfelelően!

A magyar betüket használó szövegszerkesztő mindenki számára hozzáférhető, a budapesti SINCLAIR klubban. A nyomtatók átalakitását magyar karakterkészletesre (az eredeti ASCII karakterek megtartása mellett) a MICROTEAM Számitástechnikai G.M. (1114 Ulászló u. 38. T:666-917) végzi.

Bankó Miklós

Tisztelt Szerkesztőség!

A legutóbbi szám egyik témájához szeretnék hozzászólni a következő rövid cikkel.
A lehetőség kézenfekvő, de jó sok időt eltöltöttem a -szerintem- jól olvasható és nyomtatható karakterek kialakitásával. Talán lesz, akit érdekel az ötlet.

Ékezetes betük Spectrumon

A mellékelt rövid program segitségével egyszerüen használhat-juk a teljes magyar ékezetes betükészletet.
Az 1. sorban levő pontok helyére irjuk be az 1. lista számait! RUN után a képernyőn megjelenik a billentyüzet képe a megfelelő uj karakterekkel.
Természetesen a betük elhelyezése másként is lehetséges, például a 'Character generator' programmal.
Az 1. ábrát felragaszthatjuk a gép felső részére. A kezelés rövid idő alatt megszokható.

```
  1 REM .........(168 db)
 10 LET a=USR "a"-23760
 20 FOR n=23760 TO 23927
 30 POKE n+a,PEEK n
 40 NEXT n
100 PRINT ''TAB 8;"Ékezetes bet
ük"
110 PRINT TAB 6;"Q W E R T Y U
I Ő P"
120 PRINT TAB 6;"A   É Ó Ö   Ú
Í Ő Ü"
130 PRINT TAB 7;"A S D F G H J
K L"
140 PRINT TAB 7;"á ¦ é ó ö ú ő
ü í é"
150 PRINT TAB 8;"Z X C V B N M"
160 PRINT TAB 8;"   ¦     ő ü"
170 PRINT "Az egyes betük grafi
kus üzemmód-     ban érhetők el.
"
180 STOP
200 CLEAR : SAVE "Ékezet" LINE
1: STOP
300 SAVE "Ékezet"CODE USR "a",1
68
```

Ékezetes betük

```
Q W E R T Y U I O P
A   É Ó Ö   Ú Í Ő Ü

A S D F G H J K L
á ¦ é ó ö ú ő ü í é

Z X C V B N M
  ¦     ő ü
```

Az egyes betük grafikus üzemmód-ban érhetők el.

1. ábra

```
16,0,56,4,60,68,60,0,255,0,0,0,0
,0,0,0,126,126,126,126,126,126,1
26,126,16,0,56,68,120,64,60,0,0,
126,72,64,124,64,126,0,0,40,0,56
,68,68,56,0,40,40,0,56,68,68,56,
0,40,130,130,130,130,130,124,0,0
,0,62,8,8,8,62,0,0,16,16,68,68,6
8,56,0,0,16,16,0,40,16,16,68,0,0,
0,16,56,64,68,56,0,40,40,0,68,68
,68,56,0,0,40,0,0,68,68,68,56,0,0,
60,74,66,66,66,60,0,0,40,170,130,1
30,130,130,124,0,0,60,74,66,126,
66,66,0,40,0,124,130,130,130,130,124
,0,16,16,16,16,16,16,16,16,16,40,124
,170,130,130,130,124,0,0,74,74,
66,66,66,66,60,0,
```

1. táblázat

Üdvözlettel:

Dr. Rézsa Sándor
5100 Jászberény
Riszner sétány 6

COMMODORE 64

Easy Script

Az Easy Script paragrafus-orientált szövegszerkesztő rendszer, amely szinte valamennyi mikroszámítógép-típuson rendelkezésre áll. (Az angol nyelvterületen a szófeldolgozó [wordprocessor] elnevezéssel különböztetik meg ezeket a termékeket a programozók által használt szövegszerkesztőktől. Nálunk eddig ez a terminológia terjedt el. – A szerk.)

Az Easy Script használhatóságát nagymértékben növeli, hogy beépített funkcióként az RS232, IEE 488, illetve párhuzamos (Centronics) interfészű nyomtatókat is kezeli, feltéve természetesen, hogy a gépnek van ilyen csatlakozója. A hazánkban legelterjedtebb személyi számítógépre, a Commodore 64-re is van Easy Script változat. A következőkben ezt ismertetjük. A többi változat csak apró részletekben tér el ettől.

A rendszer szolgáltatásai

Az Easy Script lehetővé teszi tetszőleges szövegek rögzítését, mágneses adathordozón való elmentését és visszatöltését. Speciális parancssorozatok helyezhetők el a szövegben, amelyek a kiírás formátumát vezérlik. A már rögzített szövegen sokféle transzformációt végezhetünk: törölhetünk, beszúrhatunk karaktereket, sorokat, megismételhetünk, áthelyezhetünk szövegrészeket. Lehetőségünk van adott karaktersorozat megkeresésére vagy cseréjére.

A szövegfájlokat összefűzhetjük; az ilyen összefűzött fájlokat az Easy Script a nyomtatás és a cserélés, keresés szempontjából egynek tekinti. A képernyőn előzetesen ellenőrizhetjük, milyen formában jelenik meg majd a nyomtatón a szöveg. Ezt külön parancsok biztosítják.

A kiírandó szöveg alapegysége a paragrafus, vagy magyar terminológiával: bekezdés. Egy bekezdést egyetlen sorként kell begépelni, azaz a <RETURN> billentyű megnyomása nélkül. A képernyőn ez természetesen több (akárhány) fizikai sort jelent.

Az Easy Script a nyomtatóra mindig úgy ír, hogy a sor végén egyetlen szót sem tör ketté. Így gyakorlatilag minden sorba felesleges szóközök kerülnek. Az Easy Script három választást kínál ezek elhelyezésére: a) a sor végén; b) a sor elején; c) a sorban egyenletesen elosztva.

A c) esetben a nyomtatón megjelenő formátum a kiszedett szövegekéhez hasonlít. A szövegben elhelyezett parancsok segítségével bármikor választani lehet az a)–c) változatok között. A fentebb mondottak alól csak a paragrafus utolsó sora képez kivételt. Ebben a sorban a szóközök vagy a sor végén, vagy a sor elején helyezkednek el.

A szöveg begépelésére a Commodore 64 szövegszerkesztőjéhez hasonló lehetőségeket nyújt az Easy Script, az idézőjel és a beszúrási üzemmód specialitásainak kivételével. Ezekre nyilvánvalóan semmi szükség.

Ha a kurzorral a képernyő alján vagy tetején túllépünk, az Easy Script a tárolt szöveget automatikusan felfelé vagy lefelé mozgatja. Két szerkesztő mód közül választhatunk. Az egyikben a nem vezérlő karakterek a kurzor helyére íródnak be. A második – inzert – üzemmódban a kurzor alatti karaktertől kezdve az egész szöveg egy karakterpozícióval hátrább tolódik. A két üzemmód közül az <F1> és I billentyűk beütésével választhatunk.

Az Easy Script a képernyő első sorát a felhasználóval való információcserére használja, így csak 24 sornyi hasznos terület van egy időben a képernyőn. Az első sor végén mindig L:xxx C:xxx alakú információt látunk, ami a kurzor sorának és oszlopának helyét jelenti. A sor elején a kiadott parancsra vonatkozó információt látjuk, és ide kell beírni a parancsok szöveges paramétereit (fájlok neve, cserélendő karaktersorozat).

A szövegben a kiírás formátumát vezérlő karaktersorozatokat helyezhetünk el. Ezeket az inverz csillag begépelésével kell kezdenünk, amit az <F3> billentyű lenyomásával érhetünk el. A következő <RETURN> lenyomásáig minden karakter a vezérlő karaktersorozat része lesz. A legfontosabb vezérlő funkciók a következők:

a) a bal és jobb margó beállítása
b) a fejléc és lábjegyzet szövegeinek megadása
c) egy lap nagyságának megadása
d) a jobb oldali szélre igazítás beállítása
e) a margóátlépés engedélyezése
f) üres sorok beírása
g) új lap kezdése

A b)-ben említett fejléc, illetve lábjegyzet három szövegrészt tartalmazhat: egyet baloldalt, egyet jobboldalt és egyet középen. Lehetőség van automatikus sorszámozás beállítására is. Az első lap sorszáma megadható. A g) alatti „új lap kezdést" úgy is használhatjuk, hogy megmondjuk, hogy csak akkor kezdjen új lapot, ha az adott lapon már nincs bizonyos mennyiségű üres sor.

Az Easy Script elsősorban nyomtatóra dolgozik, de előzőleg ellenőrizhető, hogy milyen formátumban jelenik majd meg a szöveg. A C64 a monitoron egy sorban 40 karaktert jelenít meg. A nyomtatók sorszélessége ennél lényegesen nagyobb. Így kényszerűségből az Easy Script a képernyőt ablakként ráteríti a végleges szövegre, és ezt a kurzorvezérlő billentyűkkel vagy botkormánnyal mozgathatjuk jobbra, balra és előre. A szövegben visszafelé nem olvashatunk. Ennek oka, hogy a vezérlő karakterek visszafelé nem értelmezhetők.

A szöveg nyomtatóra vagy képernyőre való kiírását az <F1> és 0 billentyűk leütésével kell kezdenünk. Ezután kell specifikálnunk a kiírás módját: melyik eszközre, folytonosan vagy laponként, a memória tartalmát vagy összefűzött fájlokat írjon-e ki, hány példányt kérünk stb.

Az Easy Script módot nyújt különböző szövegrészek törlésére, áthelyezésére, ismétlésére. Az áthelyezéshez, ismétléshez definiálni kell a mozgatandó vagy átírandó szövegrészt. Ezt a "SET RANGE" parancs segítségével lehet megadni. A parancs kiadása előtt a kurzort a megjelölendő szövegrész első karakterére állítjuk. Ezután az <F1> és R karakterek leütésével kiadjuk a "SET RANGE" parancsot, majd a kurzorbillentyűk segítségével a kurzort a megjelölendő szövegrész végére állítjuk. A kurzor mozgatása közben a kijelölt szövegrészbe kerülő karakterek inverzükre cserélődnek. A <RETURN> lenyomásával kijelöltünk egy szövegrészt.

A törlés hasonlóképpen történik: a kurzort a törlendő szövegrész elejére állítjuk, majd az <F1> és D billentyűzésével kiadjuk a "DELETE" parancsot. A kurzor mozgatásával a törölni kívánt szövegrész utolsó karakterére állunk. A törlendő karakterek inverz alakban látszanak. A <RETURN> lenyomására automatikusan végrehajtódik a törlés.

Az Easy Script rendszerben mind a teljes szöveg, mind annak kijelölt része lemezen vagy kazettán tárolható. A fájlok visszatöltése mindig a kurzor helyétől kezdődik. Ha beszúrási üzemmódban adtunk ki betöltési parancsot, a szöveg beszúródik a kurzor helyére. A fájlok neveit mindig idézőjel nélkül kell beírnunk a parancssorba.

Az Easy Script lehetővé teszi összefűzött fájlok használatát is. Ezáltal az egyidőben kiírható szövegnek csak a lemezegység tárolókapacitása szab határt, de ez sem igazán, mert a nyomtatás felfüggeszthető arra az időre, míg kicseréljük a lemezt.

A vezérlő karakterek

A vezérlő karaktereket a paragrafus elején vagy végén kell elhelyezni, sorozatukat egy in-

A kézirat kinyomtatásához a MICROTEAM Számítástechnikai GM által kifejlesztett és a magyar szabványnak megfelelő karakterkészlettel ellátott Epson FX-80 típusú nyomtatót használta a szerző

EASY SCRIPT

Az **Easy Script** olyan paragrafus-orientált szövegszerkesztő rendszer, amelyik szinte valamennyi mikroszámítógép típusnál rendelkezésre áll. Az **Easy Script** használhatóságát nagymértékben növeli, hogy beépített funkcióként a szóbanforgó számítógép perifériái mellett az RS232, IEE488 illetve a párhuzamos (CENTRONICS) interface-ü nyomtatókat is kezeli (feltéve természetesen, hogy a gépnek van ilyen csatlakozója). A hazánkban legelterjedtebb személyi számítógépre, a Commodore 64-re is létezik **Easy Script** változat. A következőkben ezt ismertetjük. A többi változat csak apró részletekben tér el ettől.

Az **Easy Script** lehetőséget biz[illegible] it tetszőleges szövegek *rögzítésére*, ezek mágneses ada[illegible]ozón való *elmentésére* illetve *visszatöl*[illegible] Speciál[illegible] hel[illegible] el[illegible] ki[illegible]

verz csillag vezeti be. Ezt az <F3> gomb leütésével tudjuk előállítani. Az egyes vezérlő utasításokat a : jellel kell elválasztani egymástól. A leírásunkban szereplő szóközöket nem szabad a szövegbe beírni, tehát "1n 2" gépelés helytelen. Helyesen "1n2". Az Easy Script a következő <RETURN>-ig mindent a vezérlő sorozat részének tekint.

A továbbiakban: x = egy- vagy kétjegyű szám, <szöveg> = tetszőleges szöveg.

lm x	bal margó beállítása
rm x	jobb margó beállítása
ma x	bal margó visszatolása (egy sorra)
ra x	a jobb margóra tömörít
pl x	egy lap fizikai sorainak száma
tl x	az egy lapra maximálisan írható sorok száma
ju x	jobb oldali szélre állítás x=Ø nincs x=1 van
hd x : <szöveg>, <szöveg>, <szöveg>	fejléc szövegei
ft x : <szöveg>, <szöveg>, <szöveg>	lábjegyzet szövegei a szövegben szereplő # (amit az <F1> <shift-#> billentyűzéssel állíthatunk elő) automatikus sorszámozást ad
p# x	a lapszámozás kezdőértékének beállítása
hl x	a fejléc és a lábjegyzet bal margója
hr x	a fejléc és a lábjegyzet jobb margója
ln x	x darab soremelés
sp x	a szövegsorok közötti soremelések száma
pt x	a nyomtatás sűrűsége x értékei 6, 8, 1Ø, 12, 15 (karakter per inch)
fp x	ha a lapon nincs már x sornak hely, akkor új lapot kezd x=Ø mindenképpen új lapot kezd
nb	megjegyzés
cn x	középre állítás x=Ø nincs x=1 van
lk: <file név>	a lánc következő eleme

A parancsok

A) Belépés parancs módba: <F1>. Ezt követően az <F2> a legközelebbi idézőjelek közötti szöveget írja be a parancssorba.

o	output
c	folytonos
l	összefűzött fájlok listázása
d	nyomtató hardver száma (4)
f	levélfájl használata
x	másolatok száma
p	nyomtatóra (printer)
v	képernyőre (video)
<C=>	következő sor
<CRSR JOBB>	jobbra
<CRSR BAL>	balra
<F5>	2Ø oszlop jobbra
<F7>	4Ø oszlop jobbra
<RETURN>	első oszlop
<szóköz>	gyors listázás ki/be
<STOP>	kilépés
c	következő oldal
v	következő oldal első 24 sora c és v csak az oldal végén használható
l	fájl betöltése a kurzor pozíciótól tölti be
	inzert üzemmód esetén beilleszti
f	fájl elmentése ha a fájlnév + karakterre végződik, akkor a tabulálási pontokat is elmenti
<SHIFT-f>	szövegrész elmentése
e	törlés
a	mindent
r	a kurzortól mindent
p	a paragrafust
s	a mondatot
d	törlés a parancs kiadása előtt a kurzort az első törlendő karakterre kell állítani, a parancs kiadása után a kurzort az utolsó törlendő karakterre kell állítani <RETURN>-re hajtódik végre a törlés
<INST>	sorbeszúrás (ismételhető), kilépés <RETURN>-nel
i	áttérés inzert üzemmódba
s	keresendő és cserélő karaktersorozat beállítása
@	csere megkezdése
m	a memóriában
l	összefűzött fájlokban
h	karaktersorozat keresése
m	a memóriában
l	összefűzött fájlokban
c	folyamatosan
g	adott sorra állás
yyy	a sorszáma
e	utolsó utáni sorra
<CRSR le>	a szöveg mozgatása lefelé
<CRSR fel>	a szöveg mozgatása felfelé
<STOP>	kilépés
<szóköz>	megállítás/újraindítás
<SHIFT>	gyorsítás
<szóköz>	következő képernyő
<SHIFT-szóköz>	előző képernyő
a	szövegrész ismétlése
x	szövegrész áthelyezése
r	szövegrész beállítása a parancs kiadása előtt a kurzort a szövegrész elejére kell állítani
	a parancs kiadása után a kurzort a szövegrész végére kell vinni, befejezés <RETURN>-nel
–	szoftver elválasztó jel
*	hangjelzés ki/be
t	tabulátor beállítása
h	vízszintes
v	függőleges a parancs kiadása előtt a kurzort a tabulálni kívánt helyre kell mozgatni
p	kijelzi a vízszintes tabulálási pontokat
c	töröl egyetlen tabulátor pozíciót
h	vízszintes
v	függőleges a parancs kiadása előtt a kurzort a törlendő pozícióra kell beállítani
z	az összes tabulálási pontot törli
x=ascii	speciális karakterek definiálása x értékei: Ø–9
x	speciális karakter beírása x értékei: Ø–9
[	duplavonalas nyomtatás kezdete
]	duplavonalas nyomtatás vége
(	inverz nyomtatás kezdete
)	inverz nyomtatás vége
:	felülírt nyomtatás kezdete/vége
'	felső index (egyetlen karakter)
,	alsó index (egyetlen karakter)

B) Belépés lemez-parancsmódba: <F4>. A parancsok megegyeznek a DOS parancsokkal. Kivételek:

$	képernyőre listázza a katalógust
+$	a szerkesztési területre is betölti a katalógust
<RETURN>	olvassa a parancs csatornát

DR. ÚRY LÁSZLÓ

ADOK – VESZEK – CSERÉLEK

Ebben a rovatban rövid, szöveges, a mikroszámítógépekkel kapcsolatos hirdetéseket közlünk. A díjszabás: közleteknek gépelt soronként (60 karakter) 100,– Ft, magánszemélyeknek az első sor 50,– Ft, minden további sor 20,– Ft. Az NJSZT tagjainak az első három sor ingyenes. Hirdetéseiket a szerkesztőség címére várjuk.

• **VEGYE IGÉNYBE AZ INTERBIT ELEKTRONIKAI GMK SZOFTVERSZOLGÁLTATÁSAIT!** Közel ötven saját fejlesztésű játék- és hobbiprogrammal állunk a vásárlók rendelkezésére, Sinclair mikrókra (1–16 k-s ZX81, 16–48 k-s Spectrum). Kérjen katalógust! Reális árak, újdonságok! Négy kazetta vásárlása esetén az ötödik ajándék! Több mint húsz felhasználói és segédprogram Sinclair és Commodore gépekre. Kihelyezett – rendszertelepítéseit! – alap- és középfokú számítástechnikai tanfolyamok szervezése. BASIC és Z80 assembly tanfolyamok. Felhasználói specifikációra programfejlesztés. Általános és középiskolás diákoknak nyári tanfolyam. Referenciahelyek. Postacím: 1631 Budapest, pf. 6. Telefon: 856–028

• **Felajánlok megrételre** új, 2 kbájtos ZX81 (Timex Sinclair 1000) számítógépet. Irányár: 10 000,– Ft. Telefon: 157-753 este.

• **AIRCOMP 16 SZEMÉLYI SZÁMÍTÓGÉP** teljes alkatrészkészlettel, összeszerelt állapotban, indítás előtt eladó. Keresek 6502 A mikroprocesszort. Takács József, Zalaegerszeg, Radnóti u. 13. 8900. Telefon: 92/13–250/353

• **COMMODORE VC–20/VIC–20-HOZ** 32 kbájtos memóriabővítő eladó. Csepi Ferenc, Iszkaszentgyörgy, Ady u. 20. 8043

• **A FOTOELEKTRONIK–NOVOTRADE GT** számítógépszervíz készséggel áll a Commodore 64 felhasználók rendelkezésére. Szervizünk rendelkezik a javításhoz szükséges anyagokkal, így többek között a 6510 típusú CPU-val is. A 6510 fogyasztói ára jelenleg 2840,– Ft.
Nyitvatartásunk: 9 órától 14 óráig. Címünk: Budapest, Magyar u. 12–14. 1053. Telefon: 173-551. Telex: 227621
Amennyiben szervizünkkel és tevékenységünkkel kapcsolatban továbbiakat szeretnének megtudni, szívesen meghívunk valakit a szerviz bemutatására.

• **COMMODORE VC–20** számítógép eladó. Zipszer Lajos, Karcag, Sallai u. 16. 5300

Backtrack algoritmus – legkisebb mélységű keresés

Sok szó esik arról, milyen ismereteket oktassunk az iskolában a számítógépekkel kapcsolatban. Többször hangoztattuk, hogy első gimnáziumban – és pár év múlva már az általános iskola felső tagozatán – minden tanulónak párbeszédes géphasználat mellett el kell sajátítania a BASIC programozás alapelemeit. Második gimnáziumtól a fakultáción kívül, haladó szakkör keretében logikai játékok, szimuláció, algoritmusok, kódolás járulhat a programozás magasabb szintű oktatásához.

Ehhez kapcsolódó elképzelésünket vázoljuk az alábbiakban. Az írást elsősorban tanároknak, esetleg idősebb szakkörvezetőknek szánjuk. Úgy gondoljuk, hogy saját kiegészítésekkel, némi didaktikával 2–5 foglalkozásnyi anyag nyerhető belőle. A bemutatott módszer megismerése azoknak is hasznos lehet, akik alaposabb programozói ismeretekre akarnak szert tenni.

1. feladat

Képzeljünk el egy igazi fedett labirintust, például egy síkbeli barlangrendszert, két kapuval a külvilág felé (*1. ábra*). A falakat felülről nem látjuk, csak azt észleljük, mehetünk-e valamerre, vagy sem. Hogyan juthatunk el a labirintus egyik végéből a másikba, vagyis a bejárattól a kijáratig?

Tegyük egyik kezünket a falra, és mindig úgy haladjunk, hogy ez a kezünk simítsa a falat. Másként fogalmazva: minden elágazásnál mindig balra (jobbra) forduljunk! Ezáltal ha nem is a legrövidebb úton, de biztosan kijutunk az útvesztőből, feltéve, hogy ez egyáltalán lehetséges. Természetesen lesznek részek, ahol kétszer is jártunk, különböző irányban, de ennél többször sehol sem fordulunk meg.

Ez a módszer a címbeli eljárás klasszikus példája. Az algoritmus igazi megismeréséhez egyszerűbb feladatokon keresztül fogunk eljutni.

2. feladat

Egy sintyó ára 8 forint, egy kafué 14 forint. 200 forint van a zsebünkben, és ezt maradéktalanul szeretnénk a fenti létfontosságú árucikkekbe fektetni.

A feladat egyetlen kétismeretlenes egyenletrendszer megoldását jelenti, és a megoldást az egész számok körében keressük. Az ilyen ún. diophantoszi egyenletek nem szerepelnek a középiskolai tananyagban, és bár elég könnyen megoldhatók, most inkább egy program segítségével próbálkozunk *(1. program)*.

Ezt az eljárást rendszeres próbálgatásnak nevezhetnénk, ami a programban egymásba skatulyázott ciklusok formájában jelenik meg. A dolog pofon egyszerűnek tűnik, de sajnos gyakran bonyolultabb a helyzet. A módszer rendszeressége komoly követelmény: fontos, hogy minden lehetőséget legfeljebb egyszer kerítsünk sorra. (A labirintusnál is egy irányban mindenütt legfeljebb egyszer jártunk!)

Ez nyilván nem túl nehéz, ha a vizsgálandó szituációk valamilyen természetes sorrendbe (nagyság szerint, ABC szerint stb.) állíthatók. Ez történt például a 2. feladatnál, az ismeretlenek (darabszámok) értékeivel. Az útvesztő esetén nem ilyen rózsás a helyzet, és több példa is van, ahol az igazi probléma éppen ez a rendezés.

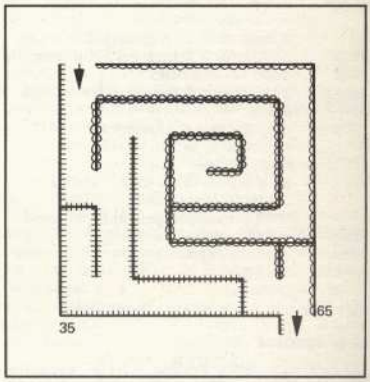
A próbálgatás céltudatossága jelentősen befolyásolja a program futási idejét, és ezáltal sok esetben a feladat megoldhatóságát is. Ez a fenti példánál is megfigyelhető: futtassuk P = 2000 értékkel is a programot! Bizony ez már 10 percig tart. Hozzáírva a

```
20 FOR J=0 TO (P-I*A)/B
```

sort, a futási idő durván felére csökken.

Az így nyert program már egy kezdetleges megvalósítása a backtrack algoritmusnak, amit magyarra legkisebb mélységű, változtatáson alapuló keresésnek szoktak fordítani. Ugyanilyen jól jellemzi az ügyes, rendszeres próbálga-

1. ábra



1. program

```
5 A=8:B=14:P=200
10 FOR I=0 TO P/A
20 FOR J=0 TO P/B
30 IF I*A+J*B=P THEN PRINTI;
   "*";A;"+";J;"*";B;"=";P
40 NEXT J
50 NEXT I
```

2. program

```
10 M=7
50 U(0)=1
100 FOR D=0 TO 9:U(1)=D
110 FOR E=0 TO 9:U(2)=E
120 FOR Y=0 TO 9:U(3)=Y
130 FOR N=0 TO 9:U(4)=N
140 FOR R=0 TO 9:U(5)=R
150 FOR O=0 TO 9:U(6)=O
160 FOR S=0 TO 9:U(7)=S
200 S1=1000*S+100*E+10*N+D:S2=1000+100*O+R*10+E
210 S3=10000+1000*O+100*N+10*E+Y
250 IF S1+S2<>S3 THEN 300
270 FOR I=0 TO 6:FOR J=I+1 TO 7:IF U(J)=U(I)
    THEN 300
280 NEXT J,I
290 FOR I= 0 TO 7:PRINTU(I);:NEXT I
300 NEXT S
310 NEXT O
320 NEXT R
330 NEXT N
340 NEXT Y
350 NEXT E
360 NEXT D
```

3. program

```
10 M1=7
50 U(0)=1
100 FOR D=0 TO 9:U(1)=D:M=1:GOTO 500
110 FOR E=0 TO 9:U(2)=E:M=2:GOTO 500
120 FOR Y=0 TO 9:U(3)=Y:M=3:GOTO 500
130 FOR N=0 TO 9:U(4)=N:M=4: GOTO 500
140 FOR R=0 TO 9:U(5)=R:M=5:GOTO 500
150 FOR O=0 TO 9:U(6)=O:M=6:GOTO 500
160 FOR S=0 TO 9:U(7)=S:M=7:GOTO 500
200 S1=1000*S+100*E+10*N+D:S2=1000+100*O+R*10+E
210 S3=10000+1000*O+100*N+10*E+Y
250 IF S1+S2<>S3 THEN 300
290 FOR I= 0 TO 7:PRINTU(I);:NEXT I
300 NEXT S
310 NEXT O
320 NEXT R
330 NEXT N
340 NEXT Y
350 NEXT E
360 NEXT D
399 END
500 FOR J=0 TO M-1:IF U(J)<>U(M) THEN 510
505 ON M GOTO 360,350,340,330,320,310,300
510 NEXT J
520 ON M GOTO 110,120,130,140,150,160,200
```

```
10 M1=7
50 U(0)=1
100 FOR D=0 TO 9:U(1)=D:M=1:GOTO 500
110 FOR E=0 TO 9:U(2)=E:M=2:GOTO 500
120 FOR Y=0 TO 9:U(3)=Y:M=3:GOTO 500
125 A=D+E-Y:B=(SGN(D+E-9.5)+1)/2:IF A*(A-10)<>0
    THEN 340
130 FOR N=0 TO 9:U(4)=N:M=4: GOTO 500
140 FOR R=0 TO 9:U(5)=R:M=5:GOTO 500
145 A=N+R-E+B:B1=(SGN(N+R-9.5)+1)/2:
    IF A*(A-10)<>0 , 320
150 FOR O=0 TO 9:U(6)=O:M=6:GOTO 500
155 A=E+O-N+B1:B2=(SGN(E+O-9.5)+1)/2:
    IF A*(A-10)<>0 , 310
160 FOR S=0 TO 9:U(7)=S:M=7:GOTO 500
200 A=S+1-O+B2:IF A<>10 THEN 300
290 FOR I= 0 TO 7:PRINTU(I);:NEXT I
300 NEXT S
310 NEXT O
320 NEXT R
330 NEXT N
340 NEXT Y
350 NEXT E
360 NEXT D
399 END
500 FOR J=0 TO M-1:IF U(J)<>U(M) THEN 510
505 ON M GOTO 360,350,340,330,320,310,300
510 NEXT J
520 ON M GOTO 110,120,125,140,145,155,200
```

4. program

```
10 M1=7
50 U(0)=1
100 FOR D=0 TO 9:U(1)=D:M=1:GOTO 500
110 FOR E=0 TO 9:U(2)=E:M=2:GOTO 500
120 A=(SGN(D+E-9.5)+1)/2:Y=D+E-10*A:U(3)=Y:M=3:
    GOTO 500
130 FOR N=0 TO 9:U(4)=N:M=4: GOTO 500
135 B=0:R=E-A-N:IF R<0 THEN R=10+R:B=1
140 U(5)=R:M=5:GOTO 500
145 C=0:O=N-E-B:IF O<0 THEN O=O+10:C=1
150 U(6)=O:M=6:GOTO 500
155 S=9+O-C:IF S>9 THEN 330
160 U(7)=S:M=7:GOTO 500
290 FOR I= 0 TO 7:PRINTU(I);:NEXT I
330 NEXT N
350 NEXT E
360 NEXT D
399 END
500 FOR J=0 TO M-1:IF U(J)<>U(M) THEN 510
505 ON M GOTO 360,350,350,330,330,330,330
510 NEXT J
520 ON M GOTO 110,120,130,135,145,155,290
```

5. program

```
90 CLS
100 INPUT "MEKKORA LEGYEN A TA'BLA OLDALA";N
480 K=2:INPUT "AZ ELS0 VEZE'R SORA AZ ELS0 OSZLOPBAN";O(1)
490 J=1:IF K=N THEN P=P+1
500 FOR I=1 TO K-1
510 IF J=O(I) THEN 550
520 IF ABS(K-I)=ABS(J-O(I)) THEN 550
530 NEXT I
535 O(K)=J
540 K=K+1:IF K<=N THEN 490 ELSE GOSUB 1000
550 J=J+1:IF J<=N THEN 500
555 K=K-1:IF K=1 THEN IF S=0 THEN 480 ELSE END
560 J=O(K):GOTO 550
1000 FOR J=1 TO N:PRINTCHR$(64+J);CHR$(48+O(J));" ";:NEXT J
1010 S=S+1:PRINT,:RETURN
```

6. program

tás elnevezés is. Az ügyességre vonatkozóan nézzünk egy még tanulságosabb esetet.

3. feladat

Képzeljük el, hogy a jól ismert tízes számjegyjelölések (0...9) helyett betűkkel jelöljük az egyes számjegyeket. Nem tudunk semmit az egyes betűknek megfelelő számjegyek értékéről, csak azt tudjuk, hogy az alábbi összeadás igaz:

```
  SEND
+ MORE
 MONEY
```

(Ez a jól ismert betűszámtan-feladat egyébként angolul azt jelenti: KÜLDJ TÖBB PÉNZT.)

Elöljáróban vegyük észre a triviális M=1 összefüggést. Ez nyilvánvalóan adódik, hiszen két bármilyen értékű tízes számjegy összeadásakor csak 1-es értékű átvitel keletkezhet. Ezáltal egy hétismeretlenes rendszerhez jutunk, négy egyenlettel, és néhány kiegészítő feltétellel: minden betű helyén más egész szám áll.

A 2. feladathoz hasonlóan hét darab egymásba skatulyázott FOR ciklussal próbálkozunk, és legbelül vizsgáljuk az összeadási feltételt, illetve a különbözőséget *(2. program)*.

Hiába egyszerű a programunk, sajnos futási ideje jócskán szétfeszíti az iskolai lehetőségek kereteit. Ráadásul erről nem az iskolaszámítógép tehet, jobb személyi számítógépeken is több nap alatt adódik a megoldás. Próbáljuk célratörőbbé tenni a vizsgálatot! *(3. program)*

A program 6–7 óra alatt véget is ér. Ez nem meglepő, hiszen 10^7 összeadás-vizsgálat helyett „csak" $9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \sim 2 \cdot 10^5$ szerepel. A különbözőségeket igyekeztünk minél korábban tesztelni, és ha valami nem ment, azt nem erőltettük tovább. Ha a harmadik változó nem volt megfelelő, akkor a 4. és a többi már szóba sem jött. Erre utal a legkisebb mélység elnevezés. Megyünk, amíg lehet, ha nem tudunk továbblépni, akkor az utolsó lépést változtatjuk meg, ha lehet; ha nem, akkor az előzőt stb.

Az idő jelentős rövidülésének ára mindössze egy kis többlet: a 100 és a 160 közötti sorok módosítása, és az ON utasítások elágazásainak alapos végiggondolása.

Ezen felbuzdulva próbáljuk meg az összegek vizsgálatát is előbbrehozni: végezzük azt jegyenként. Ehhez az átvitel (marad 1 vagy sem) kérdését kell átgondolni *(4. program)*.

B, B1, B2 jelentik az esetleges átvitelt rendre jobbról az 1., 2., 3. oszlopban: értékük 1, ha marad 1, különben 0 (125, 145, 155-ös sorok). A 4. oszlopban tudjuk, hogy van átvitel, ezt S-hez hozzá is adjuk a 200-as sorban. Fantasztikus dolog történik: a futási idő lényegében 10 perc lesz!

Tulajdonképpen meg is állhatunk, hiszen a legkritikusabb szemlélő is elégedett lehet, különösen az előzmények ismeretében. A teljesség kedvéért azonban javasolunk még egy módosítást.

Vegyük észre, hogy az első oszlopban E és D értéke szinte meghatározza Y-t, csak az átvitelben vagyunk bizonytalanok. Felesleges tehát Y-t is FOR ciklussal keresni, és ugyanez vonatkozik az R, O, S változókra is. Drasztikusan csökkentve a ciklusok számát, az *5. programot* kapjuk.

Az eredmény már alig érdekes: a futási idő kb. másfél perc.

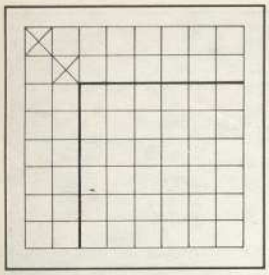
Azonnal le kell szögezni: senkitől sem kívánjuk, hogy mindjárt rájöjjön az utolsó változatra. A folyamat a lényeges: egy rutinszerű, mechanikus megoldásból ötletteli program született, és mindez csak azért, mert a számítógép lassú. A négy változat közül a harmadik és a negyedik mindenképpen magán viseli a backtrack jegyeit, a másodiknál ez egy kicsit vitatható.

Az algoritmus pontos megfogalmazása előtt nézzünk még egy klasszikus problémát.

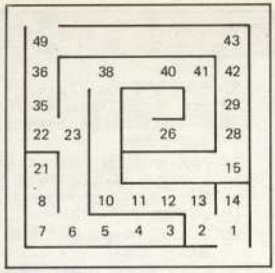
4. feladat

Helyezzünk el egy sakktáblára minél több királynőt úgy, hogy azok ne üssék egymást, azaz egyetlen sorban, oszlopban, illetve átlóval párhuzamos vonalban ne álljon több figura. (A nyolc királynő problémájával egyébként lapunk 1984/5. számában már foglalkoztunk.)

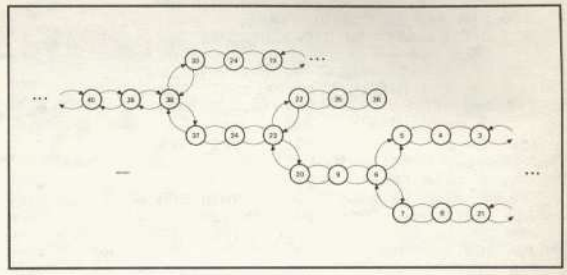
Természetesen ismét kedvenc algoritmusunkkal próbálkozunk, de azonnal érezzük, baj van a rendezéssel. Az triviális, hogy nyolcnál



2. ábra



3. ábra



4. ábra

több vezér nem lehet (minden sorban legfeljebb egy), de ez is $8! \sim 40$ ezer „állás" kiértékelését jelentené, és itt semmilyen rendezés nem adódik.

Szerencsére a tábla vonalainak jelölése (A–H és 1–8) segít. Említettük, hogy minden oszlopba legfeljebb egy vezér kerülhet, és optimista módon meg is próbáljuk a 8 királynőt felrakni a táblára. Minden vezért igyekszünk minél közelebb tenni a tábla aljához. Ha egy oszlopba nem tudunk már rakni, akkor a közvetlenül megelőző oszlopban módosítunk (felfelé toljuk azt a királynőt egy sorral). Erről az optimista változatról kapta az eljárás a nevét: mindig a legkisebb mélységben próbálunk változtatni. BASIC-re fordítva: mindig az utolsónak megnyitott, legbelső ciklus változóját igyekszünk növelni.

A sorok (0–7 az A–H helyett) és oszlopok (1–8) megjelölése révén minden állásnak megfeleltetünk egy nyolcjegyű, nyolcas számrendszerbeli oktális számot: a szám k-adik jegye azt mutatja, hogy a k-adik oszlop hányadik sorában áll vezér. Ennek megfelelően az ütés tiltása:

– egyetlen oktális szám se tartalmazzon azonos jegyeket (ütés sor mentén),

– bármely két jegy különbsége térjen el pozícióik (helyiértékük sorszáma) különbségétől, abszolút értékben. Például 02461357 elrendezésben az első és utolsó jeggyel van baj: 7–0=8–1; átlósan ütik is egymást a megfelelő vezérek.

Következő programunk egy kissé általánosabb feladatot old meg: NxN-es sakktáblán helyezi el a vezéreket, sőt az első királynőt tetszés szerint tehetjük az első oszlop bármelyik mezejére. 5–9 oldalhosszúságú táblával érdemes kísérletezni. Kisebb tábla érdektelen, nagyobb esetén pedig igen sokáig tart a program futása *(6. program)*.

A program első látásra kicsit bizarr: lényegében egyetlen FOR ciklust tartalmaz. Rejlik benne egy másik ciklus is, IF-fel szervezve (550-es sor). Ez a módszer elég gyakori hasonló feladatoknál: K ciklusváltozót hol felfelé, hol lefelé léptetjük egyesével. A program az imént megfogalmazott eljárás szó szerinti fordítása: a K-adik vezért a K-adik oszlop O(K)=J-edik sorába próbáljuk tenni, az esetleges ütéseket az 510–520-as sorok vizsgálják. Az 555-ös sorban realizálódik az előző oszlopban történt módosítás.

Ezen a példán is jól megfigyelhető az algoritmus céltudatossága. Ha egy állásban egy hiba már van, nem keres többet, hanem egy egész álláscsoportot elvet (lásd a *2. ábrát*), a hiba 6! lehetőséget ejt ki azonnal.

A 490 J=1: IF K=N THEN P=P+1 módosítás leszámolja, hányszor próbálkoztunk egyáltalán az utolsó oszlopban, azaz hány állást néztünk végig 8! helyett.

Ezek után megpróbálhatjuk megfogalmazni a backtrack algoritmus lényegét.

Az eljárás célja: egy halmazból bizonyos tulajdonságcsoportnak eleget tevő elemeket választunk ki. Az elemek a feladatokban: útvonal, darabszám, számjegyek, állás a sakktáblán.

Valamilyen rendezéssel biztosítjuk, hogy minden lehetőség legfeljebb egyszer kerüljön sorra a válogatás (tesztelés) során. A rendezések: jobbra tartás következetesen, természetes növekvő sorrend, növekvő sorrend, sorok szerint növekvő sorrend.

Igyekszünk minél előbb minél több vizsgálatot elvégezni. Ha egyszer hibára bukkanunk, azonnal módosítunk. Ezért vitatható a 3. feladat második megoldása: itt csak a különbözőséget vizsgáljuk azonnal, az összeg tesztelése a végére marad (vö. harmadik megoldás). A módosítás optimista módon, minél kevésbé visszanyúlva történik.

Felsorolunk néhány további területet, ahol a fenti algoritmus hasznos lehet.

1. Permutációk, kombinációk, partíciók és egyéb, a kombinatorikában előforduló halmazok generálása.

2. Más betűszámtan feladatok:

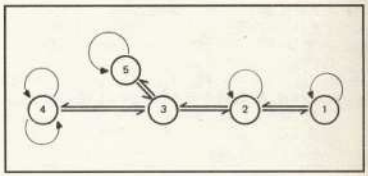
	ABB	
ÖT	BAB	KLEIN
+ÖT	+BBA	+ VIETA
TÍZ	BBBO	NEWTON

Ezek és továbbiak J. Lehmann: Furfangos matematika (Bp. 1976. Gondolat) című könyvében találhatók.

3. Bizonyos egész értékű optimalizációs feladatok, például egyszerűbb logikai játékok nyerő stratégiája (TIC–TAC–TOE, kirakósok, awari(maya), esetleg 15-ös játék stb.).

4. Speciális út (kör, legrövidebb út, egy vonallal való megrajzolhatóság stb.) keresése gráfokon. Ezúttal először rendezést kell definiálni, ami a csomópontok megjelölésével (szám, betű stb.) történik. Ezáltal N csomópont és K él esetén N-es számrendszerbeli, K jegyű számok vizsgálatára vezethető vissza a probléma. Sajnos ez még közepes méretű gráfok esetén is óriási feladat ($\sim N^K$ eset), így nem meglepő, hogy profik sokszor egész más, jóval bonyolultabb algoritmusokat használnak.

5. Megjegyezzük, hogy az útvesztő-probléma rokonságot mutat a gráfok vizsgálatával. Tekintsük a bevezetőben említett labirintust *(3. ábra)*. Megszámozva a mezőket (a folyamatos számozás az ábrán a 14. mező után megszűnik), egy irányított gráfhoz *(4. ábra)* jutunk, ahol két mezőnek megfelelő pont duplán össze van kötve, ha érintkeznek.



5. ábra

Ha csak az elágazási pontokat számozzuk (1→1, 38→2, 23→3, 6→4, 49→5), akkor sokkal tömörebb gráfot nyerünk *(5. ábra)*.

Lefordítva „gráf nyelvre" az útvesztő-problémát: tetszőleges pontból kell egy pontba jutni. Ezúttal két esetet kell megkülönböztetni: ismerjük-e a gráfot, vagy tényleg bent vagyunk-e a labirintusban. Az utóbbi esetben marad a jobbra (balra) tartás a gráfon:

5–3–4–4–4–3–2–1 (5–5–3–2–2–1)

Ha felülről is ránézhetünk az útvesztőre, vagyis ismerjük a gráfot, akkor törekedhetünk a legrövidebb útra: a hurkokat például eleve kerüljük. Ebben az esetben választ adhatunk elméleti kérdésekre is: áthatolható útvesztő → összefüggőség vizsgálata; a labirintus több értelmű → kör létezése. Természetesen a kérdéseket is pontosabban kell megfogalmazni, de ezt az érdeklődőbbekre bízzuk.

Elsődleges célunk egy igen hasznos algoritmus ismertetése volt. Mindenki tudja, mennyire fontos ez a fogalom a programozásban. Igyekeztünk úgy fogalmazni, hogy minél általánosabb érvényű legyen. A főbb gondolatokat így foglalhatjuk össze:

- A feladatok megoldása általában nem egyértelmű, több különböző program is lehet a végeredmény. Hatásfokuk esetleg jócskán eltér.
- A legszárazabb „mechanikus" eljárásban is rengeteg heurisztika ötvözhető. Ez adja a programozás igazi szépségét, ami általában hasznossággal is párosul.
- Ügyancsak művészet megtalálni az optimumot a program áttekinthetősége (bonyolultsága) és hatékonysága között.
- A programozni tudás eszköz, nem cél. Sokkal fontosabb valami egyéb tudás, nevezetesen az algoritmizálható gondolkodás.

Végül két könyvre hívjuk fel az érdeklődők figyelmét: Lovász–Gács: Algoritmusok (Bp. 1978. Műszaki Könyvkiadó) és J. Nievergelt–J. C. Farrar–E. M. Reingold: Matematikai problémák megoldásainak számítógépes módszerei (Bp. 1977. Műszaki Könyvkiadó). Ez utóbbi foglalkozik a backtrack algoritmussal.

TÖRÖK TURUL

A Proper–16 személyi számítógép

A Proper–16 a Számítástechnikai Koordinációs Intézet 16 bites, IBM PC kompatibilis professzionális mikroszámítógépe. Kifejlesztését az intézet Hardware Rendszertechnikai Laboratóriuma (HRL) végezte, a sorozatgyártást és forgalmazást pedig a Sci-L (az SZKI leányvállalata) végzi.

Csúcsmodell

A Proper–16 a legnagyobb teljesítményű, sorozatban gyártott magyar számítógép. Eddig több mint 200 konfiguráció készült.

Központi egységének sebessége és 1 Mbájtos memóriacímzési képessége alapján a gép lehetőségei sokkal inkább hasonlíthatók a nálunk forgalomban levő 16 bites miniszámítógépekéhez (például SZM4, TPA–1148), mint a 8 bites mikrogépek adottságaihoz.

Hozzávéve ehhez az IBM kompatibilitásból adódó, kiemelkedően jó szoftverellátottságot és a gép megbízható működését, a kb. egyéves felhasználói tapasztalatok alapján a gépet – kategóriájától függetlenül – a Kelet-Európában gyártott legjobb számítógépek közé sorolhatjuk. Különösen akkor, ha néhány bántó, de könnyen kiküszöbölhető hibáját (lásd alább) megszüntetnék.

A központi egység

A gép központi egysége az Intel 8088-cal kompatibilis. Jellegzetessége a 16 bites belső architektúra és a 8 bit szélességű külső adatsín (perifériabusz). Címtartománya 20 bites, mely 1 Mbájtos központi tár használatát teszi lehetővé. A tár nem lineárisan címezhető, hanem 64 kbájt méretű szegmensekre oszlik. A címregiszterek általában 16 bitesek; a kívánt lap szegmensregiszter segítségével választható ki.

Mivel a program szegmensei különböző memórialapokat is használhatnak, a Z80-tól (vagy a TPA–1140-től) eltérően egy program mérete nincs 64 kbájtban korlátozva.

A gép központi egysége egy négyrétegű nyomtatott áramköri kártyán helyezkedik el, amelyen 48 kbájt ROM és max. 256 kbájt RAM memória is helyet kapott.

A ROM egy része tartalmazza a gép felhasználását elősegítő mikroprogramot (BIOS), 40 kbájtba pedig egy BASIC interpreter van beégetve, amely háttértár nélkül is használható. A központi egység órafrekvenciája 4,77 MHz. A kártyán bővítésként elhelyezhető az Intel 8087 típusú lebegőpontos segédprocesszor is, amely a gépet különösen alkalmassá teszi műszaki-tudományos számítások végzésére. Egy lebegőpontos szorzás átlagos időszükséglete kb. 20 μs.

Típus	Formátumozatlan kapacitás	Formátumozott kapacitás	Átviteli sebesség
Hajlékony mágneslemez (floppy) meghajtó 1. típus	500 kbájt	320 kbájt (8 szektor/sáv)	250 kbit/s
		360 kbájt (9 szektor/sáv)	
Hajlékony mágneslemez (floppy) meghajtó 2. típus	1 Mbájt	720 kbájt (9 szektor/sáv)	250 kbit/s
Merevlemez (Winchester) meghajtó 1. típus	9,57 Mbájt	7,78 Mbájt	5 Mbit/s
Merevlemez (Winchester) meghajtó 2. típus	27,5 Mbájt	22,39 Mbájt	5 Mbit/s

1. táblázat. A Proper-16-hoz forgalmazott mágneslemez-meghajtók teljesítményadatai

Az alaplemezen elhelyezkednek el a max. 8 db bővítőkártya beépítését lehetővé tevő, 64 pólusú Európa-csatlakozók is.

A bővítőkártyák

A kisebb, Proper–16/A jelzéssel forgalmazott konfigurációhoz legalább két bővítőkártya szükséges: a kijelzővezérlő és a hajlékonylemez-vezérlő.

A *kijelzővezérlő* ún. „composite video" és szabványos RGB (Red-Green-Blue) kimenettel egyaránt rendelkezik, és mind monokromatikus, mind színes monitor csatlakoztatható hozzá kijelzőként. A karakterek 5 × 7 méretű pontmátrixban ábrázolhatók, a képernyőn 25 sor és soronként 80 karakter jeleníthető meg.

A kártya a Motorola 6845-tel kompatibilis vezérlő áramkört használja; ez lehetőséget nyújt raszter típusú grafika megjelenítésére is. A felbontás monokromatikus (zöld foszforos) kép esetén 640 × 200, 16 szín használata esetén 320 × 200 képpont.

A *hajlékonylemez-vezérlő* a tipikus konfigurációkban két darab 5 1/4"-es hajlékonylemezes tárolót hajt meg. Az első gépeket 140 kbájt kapacitású MOM lemezegységekkel szállították, később pedig 360, illetve 720 kbájt kapacitású BASF egységeket építettek be. Az utóbbi esetben – 4 lemezegység használatával – a hát-

tértár-kapacitás megközelíti a 3 Mbájtot, és megőrizhető a kompatibilitás az IBM PC-n felírt 360 kbájtos lemezekkel is.

Szinte minden konfiguráció tartalmazza a *nyomtatóvezérlőt*, amelyhez a gyártó az elmúlt évben saját fejlesztésű MP–80 mátrixnyomtatóját értékesítette, illetve ITOH–8510 vagy Terta TMT–120 nyomtatót szállított. Ugyanez a kártya tartalmaz egy aszinkron vonali illesztőt is, amely lehetővé teszi a Proper–16 összekapcsolását más számítógépekkel soros (V.24) vonalon, max. 9600 Baud sebességig.

A *merevlemezes háttértárvezérlő* segítségével a géphez 5 1/4"-es Winchester-lemez illeszthető, amelyhez jelenleg 7 vagy 22 Mbájtos meghajtókat szállítanak.

Az illesztő egy Z80 processzorra épülő önálló mikrogép, amely a lemezegység vezérlésére korszerű, nagy integráltságú LSI áramkört használ.

Az így létrehozható, Proper–16/W típusként forgalmazott konfigurációk a felhasználó számára teljesen kompatibilisek a világszerte széles körben elterjedt, IBM PC/XT jelzésű mikrogéppel. A hajlékonylemez- és Winchester-lemez meghajtók adatait az *1. táblázatban* foglaltuk össze.

Ebben a kategóriában természetesen felmerül a központi *tár bővítésének* igénye is. Ezt egy max. 512 kbájt RAM memóriát tartalmazó kártya teszi lehetővé.

A Proper–16 hardver-építőelemet a *2. táblázat* mutatja.

Az alapszoftver

A Proper–16 operációs rendszere a PROPOS–16. Ennek 3. verziója felülről kompatibilis a Microsoft MS–DOS (a Microsoft Corp. védjegye) rendszerének 2.0 verziójával. Ez az operációs rendszer fő vonásait tekintve hasonló a CP/M-hez. A PROPOS–16 legfontosabb parancsai:

ASSIGN	bemenet/kimenet átirányítás
CHDIR	új directory beállítása
COMP	állományok összehasonlítása
COPY	állománymásolás
DATE	dátumbeállítás
DIR	tartalomjegyzék-készítés
DISKCOMP	lemez-összehasonlítás
DISKCOPY	lemezmásolás
ERASE	állománytörlés
FORMAT	lemezformattálás
MKDIR	új directory készítés
MODE	vonali paraméterek beállítása
PRINT	nyomtató spooler
RENAME	állomány átnevezése
RMDIR	directory törlés
TIME	időbeállítás
FSYS	teljes directory struktúra kiíratás
TYPE	állomány kiíratása
VERIFY	automatikus ellenőrzés lemez be/kiviteli műveleteknél

Ez a rendszer történeti okok miatt is egységesebbnek, a professzionális célú felhasználásra jobban alkalmazhatónak tűnik, mint a CP/M. Érdemes azonban megemlíteni, hogy a CP/M-et kifejlesztő Digital Research cég CONCURRENT DOS 3.1 néven 1984-ben megkezdte egy

Megnevezés	Ár
Proper–16 alapegység: – központi tár (256 kbájt RAM, 40 kbájt ROM, billentyűzetcsatolás, hangszóróilleszték, tápegység, készülékház 8 bővítési kártyahellyel, standard grafika), – billentyűzet.	420 000 Ft
Lebegőpontos aritmetikai modul	49 000 Ft
128 kbájt RAM bővítőkártya	60 000 Ft
256 kbájt RAM bővítőkártya	90 000 Ft
512 kbájt RAM bővítőkártya	150 000 Ft
Kijelzőcsatoló (színes vagy monokromatikus monitorhoz)	40 000 Ft
Soros vonali csatoló (2 db V24 szinkron vagy aszinkron vonal)	35 000 Ft
Hajlékonylemez-csatoló (4 db 5,25"-os vagy 8"-os szimpla vagy dupla írássűrűségű lemez illesztésére)	27 000 Ft
Nyomtatócsatoló (párhuzamos interfész)	37 000 Ft
Grafikus csatoló* (512 × 512 képpont felbontás, 16 kbájt tároló)	55 000 Ft
Merevlemezes tárolócsatoló (4 × 5,25"-os merevlemezes tár illesztésére)	95 000 Ft
U880D adaptercsatoló* (Z80 programok futtatására)	50 000 Ft
Lokális hálózati csatoló*	95 000 Ft
Csoportos terminálvezérlő (master)*	40 000 Ft
Csoportos terminálvezérlő (slave)* (6 terminál csatlakoztatására)	45 000 Ft
Színes professzionális monitor	145 000 Ft
Monokromatikus professzionális monitor	30 000 Ft
MP–80 mátrixnyomtató	88 000 Ft
5,25"-os hajlékonylemez-meghajtó (BASF 6128, kétoldalas 40 trackes)	29 000 Ft
5,25"-os hajlékonylemez-meghajtó (BASF 6138, kétoldalas 80 trackes)	35 000 Ft
2 × 8"-os hajlékonylemez-meghajtó (MF–6400)	155 000 Ft
5,25"-os 7 Mbájt kapacitású merevlemezes tároló (BASF 6138)	259 000 Ft
5,25"-os 22 Mbájt kapacitású merevlemezes tároló (BASF 6185)	300 000 Ft
AM–12TD modem	70 000 Ft
TELETERM terminál kiépítéstől függően	30–108 000 Ft
TELEDATA terminál (tájékoztató ár)	70 000 Ft
Aszinkron terminál (tájékoztató ár)	79 000 Ft

* 1985 második felétől szállítják

2. táblázat. A Proper–16 professzionális személyi számítógép hardverelemei (1985. február 1-én érvényes árak)

új operációs rendszer forgalmazását, amely a CP/M és MS–DOS szolgáltatásait egyaránt tartalmazza, ráadásul egyidőben több feladat párhuzamos végrehajtására is alkalmas.

Mivel az elmúlt években az IBM PC és a vele kompatibilis rendszerek értékesítése a mikrogépes világpiac közel felét jelentette, e gépekre ma egyedülállóan nagy szoftverkészlet áll rendelkezésre. Ez magában foglalja a CP/M alatt kifejlesztett fontos programcsomagok jó részének 16 bites változatát, ezenfelül azonban lehetővé vált olyan új típusú integrált programrendszerek kifejlesztése is, amelyek a hardver korlátai miatt a 8 bites gépeken racionális módon nem valósíthatók meg. Ezek tipikus elemei: a szövegszerkesztő (wordprocessor), az elektronikus megoldólap (spreadsheet), a grafikus programcsomag és a relációs adatbázis-kezelő rendszer. Felhasználói felületük egységes, és az egyes komponensek adatformátuma kompatibilis, így egymás eredményeit közvetlenül feldolgozhatják.

Néhány jellegzetes, a Proper–16-hoz forgalmazott programcsomag főbb adatait a *3. táblázat* tartalmazza.

Többfelhasználós rendszerek

A Winchester-lemezekkel rendelkező Proper–16 konfigurációk alkalmasak több felhasználó egyidejű kiszolgálására is. 1985 második felétől az SZKI két operációs rendszer alatt is támogatni fogja ezt az üzemmódot.

Az egyik változat háromfelhasználós. Nagymértékben kompatibilis az egyfelhasználós alaprendszerrel; a PROPOS–16 alatt működő alkalmazói programok megfelelően konfigurálva ez alatt is futtathatók. Ez a változat elsősorban nem programfejlesztésre, hanem programfuttatás céljára szolgál.

A Proper–16 másik többfelhasználós operációs rendszere ezzel szemben éppen a programfejlesztés terén biztosít a hazai mikrogéppiacon egyedülálló lehetőséget. Kompatibilis a 16 és 32 bites mikro- és miniszámítógépek között széles körben elterjedt Unix (a Bell Laboratórium védjegye) operációs rendszer V.7 változatával. Az SZKI tájékoztatása szerint még ez év végéig megkezdődik a Unix fejlettebb, System III változatával kompatibilis operációs rendszer forgalmazása is.

A Unix-kompatibilis rendszerek legfontosabb jellegzetessége a hordozhatóság: a felhasználó számára a rendszer a központi egység típusától függetlenül azonos környezetet nyújt, és a kifejlesztett programok bármelyik gépen felhasználhatók. E rendszerek kiemelten támogatott nyelve a C programozási nyelv, amelyben maga az operációs rendszer is készült. A C leginkább a PASCAL-hoz hasonlítható, strukturált programozási nyelv. Mind a C nyelv, mind a belőle fordítással előállítható bináris kód rendkívül tömör és hatékony.

Természetesen rendelkezésre állnak a más rendszerek alatt megszokott magas szintű nyelvek (COBOL, FORTRAN, PASCAL) fordítóprogramjai is.

A rendszer a kiemelkedően jó programfejlesztési eszközökön kívül igen könnyen használható, ún. menü-orientált felhasználói felüle-

Programcsomag	Ár
EXTRABASIC–16 BASIC interpreter	35 000 Ft
BASIC–16 fordítóprogram	30 000 Ft
PASCAL–16 fordítóprogram	25 000 Ft
FORTRAN–16 fordítóprogram	20 000 Ft
C–16 fordítóprogram*	30 000 Ft
PL/1–16 fordítóprogram*	30 000 Ft
COBOL–16 fordítóprogram*	30 000 Ft
FORTH–16 interpreter	20 000 Ft
MPROLOG–16 PROLOG nyelvi rendszer*	70 000 Ft
SORVAL–16 rendező-válogató program	20 000 Ft
PROP–F16 szekvenciális indexelt és relatív állománykezelő	80 000 Ft
PERDATIN–16 adatbeviteli rendszer	50 000 Ft
PROPBASE–16 relációs adatbázis-kezelő	100 000 Ft
PROPVT52–16 terminál emulátor TPA gépekhez	50 000 Ft
PROPERJE–16 IBM RJE emulátor	60 000 Ft
PROPDIAL–16 IBM 327X emulátor	70 000 Ft
PROPHSZR–MICRO–16 hálótervezési programcsomag	80 000 Ft
PROP–TEXTMAN szövegfeldolgozó rendszer	40 000 Ft
PROP–PERSTAT–16 statisztikai programcsomag	150 000 Ft
PROP–MATSTAT–16 matematikai programcsomag (8 db különálló programcsomag) egyenként	16 000– 80 000 Ft
PROP–VOISOFT beszédfeldolgozás és szintézis	
PROP–GRETA–16 interaktív képszerkesztő	60 000 Ft
PROP–LABSOFT kémiai programcsomag	
PROP–SCALC–16 elektronikus munkalap	50 000 Ft

* Fejlesztés alatt

3. táblázat. A Proper–16 számítógéppel forgalmazott néhány fontosabb programcsomag

tet biztosít az alkalmazói programrendszerek többfelhasználós környezetben történő üzemeltetéséhez.

A Unix kompatíbilis rendszerekhez képernyő-orientált szövegfeldolgozó rendszer és többfelhasználós relációs adatbázis-kezelő rendszer is beszerezhető lesz.

Értékesítés és vevőszolgálat

A Proper–16 gépekkel kapcsolatos vevőszolgálati tevékenységet az SZKI leányvállalata, a Sci-L végzi. Kérésre a gépeket, az alkalmazói szoftvereket élő bemutatókon is ismertetik. Tájékoztatással, tanácsadással, számítógépesítési tanulmányok készítésével, egyedi rendszerek beszerzésével és megvalósításával állnak a felhasználók rendelkezésére. A szoftver- és hardvertermékekkel kapcsolatos oktatást az SZKI megbízásából a SZÁMALK végzi. A termékekhez jó minőségű dokumentációt adnak – bár előfordult, hogy a felhasználói kézikönyveket csak késve kapták meg a vásárlók.

Az eddig leggyakrabban értékesített típuskonfigurációk a következők voltak.

Proper–16/A típuskonfiguráció: Proper–16 alapegység, hajlékonylemez-csatoló, nyomtatócsatoló, 2 db 5,25"-os BASF 6128-as hajlékonylemez-egység, kijelzőcsatoló, monokróm monitor, PROPOS–16, EXTBASIC–16, PASCAL–16, SORVAL–16, PERDATIN–16, TEXTMAN–16. Ára: 842 000,–Ft.

Proper–16/W típuskonfiguráció: ugyanaz, mint fent, de kiegészítve 1 db 5,25"-os, 7 Mbájt kapacitású merevlemezes (Winchester) tárolóval, a szükséges csatolóval, valamint PERINFO–16 alkalmazásgenerátorral. Ára: 1 326 000,–Ft.

A Proper–16 bérletezését a Scitel – az SZKI másik leányvállalata – és a Sci-L is végzi. Ennek formája és árai nagyjából megfelelnek a hazai gyakorlatnak. A Sci-L-től kapott tájékoztatás szerint egy Proper–16/A bérleti díja havonta – a bérleti időtől és a konfiguráció kiépítésétől függően – 30-42 ezer Ft között van. Programokat nem bérleteznek, azokat meg kell vásárolni.

A gépet folyamatos bérletezés után csökkentett áron meg lehet vásárolni: 2 év után az aktuális ár 80, 3 év után 40, 4 év után 10 százalékáért adják el.

A Sci-L a szállított gépekre 1 év garanciát vállal. A javítást a hiba bejelentésétől számított 2 munkanapon belül megkezdik. Az eladott gépek számának növekedésével az utóbbi időben ez sajnos nem mindig teljesült. A Sci-L értesítette a felhasználókat, hogy a Proper–16 szervizelését 1985-től kezdődően az Agro-Industria Innovációs Vállalat SKILCO PC szervize veszi át. Így várható, hogy ez a probléma rendeződik. A gépek garancián túli szervizét is ez a cég fogja ellátni. Tájékoztatásuk szerint a garanciaidőn túli javításokra átalánydíjas szerződés is köthető.

Szépséghibák

A Proper–16 esélytelenül indulna a „legszebb magyar számítógép" címért rendezett versenyen.

Ez persze nem tragédia, de nem is öröm a felhasználó számára. Persze tudjuk, hogy a tartalom fontosabb, mint a forma. Mégis, van egy pont: a kijelző (monitor), ahol a formai igénytelenség már a gép használhatóságát kezdi gátolni. Nem világos, hogy a rossz képminőségért elsősorban az Orion gyártmányú monitor vagy az illesztőkártya, esetleg ezek csatlakoztatása-e a felelős, a végeredményt tekintve azonban ez igen nagy kár egy szövegfeldolgozásra és grafikára egyébként ennyire kitűnően alkalmazható gépnél. (A kipróbált példányok egy részén az „inverz video" üzemmódban megjelenített szöveg gyakorlatilag nem volt olvasható.) Az SZKI tájékoztatása szerint 1985-ben megkezdik egy nagyobb felbontású, monokróm kijelző csatolókártya forgalmazását, mely várhatóan jobb képminőséget biztosít majd.

A billentyűzeten igen kellemes a NUM LOCK és a CAPS LOCK gombok jelenléte, igen kellemetlen azonban, hogy nem tudni, hogy ezek milyen állapotban vannak. Ezt ugyanis se lámpa ki nem jelzi, se a gomb helyzete nem kétállású. Megesett, hogy emiatt egy számjegy leütése helyett félórai szövegszerkesztésem eredményét tettem tönkre. Ez a probléma azonban az IBM PC és PC/XT modellek billentyűzeténél is fennáll.

A géphez készült egy olyan illesztőkártya, amelyen egy soros és egy párhuzamos vonal van, de sajnos a párhuzamos vonal egyirányú, mivel csak nyomtató támogatására szolgál.

Miután az illesztőkártya szinte üres, nem jelentett volna nehézséget a nagyobb sebességű, kétirányú kapcsolatot lehetővé tevő, párhuzamos felület kialakítása, mely sok felhasználáshoz elengedhetetlen.

Összefoglalás

A Proper–16 (és különösen a Proper–16/W) sorozat gyártásával ugyanúgy új korszak kezdődött a magyar számítástechnikában, mint az elmúlt évtizedben a TPA–11/40 bevezetésével; ezzel a termékkel a magyar számítástechnikai ipar egy „de facto" világszabványt, az IBM PC-t, honosított meg.

Nem érdemtelenül kapta meg a Proper–16 az 1984. évi BNV nagydíját. Külön kellemes meglepetés a gép nyugat-európai színvonalú megbízhatósága. Némi összerázódás után – mely arra utal, hogy a végbemérés alapossága még nem felülmúlhatatlan – az általam használt gépek üzembiztonsága megközelítette a jó nyugati termékekét, a meghibásodásokat pedig a vevőszolgálat viszonylag hamar kiküszöbölte.

Be kell vallanom, hogy az első példányokkal kiszállított MOM 5,25"-os hajlékonylemezes egységekre a megbízhatóságra vonatkozó előbbi kijelentés messze nem igaz. Körülbelül 20 egységgel szerzett tapasztalataim alapján ezek forgalmazását jóindulatú félreértésnek kell tekintenem.

A Proper–16 felhasználói olyan gép birtokába jutnak, amely megfelel a 80-as évek világszínvonalának. Reméljük, hogy a jövőben a hazai piacon sok ilyen színvonalú gép jelenik meg, és érdeklődéssel várjuk a SZKI Proper családjának újabb tagjait.

BRUCK PÉTER

Alapozás IX.

Pneumatikus operátoraink „kimeneti oldalán" olyan jelenségeket, olyan folyamatokat tapasztalunk, amelyek az operátorok „bemeneti oldalán" zajló jelenségektől (folyamatoktól) függnek. A bemeneti folyamatok (jelenségek) határozzák meg, alakítják ki a kimeneti folyamatokat (jelenségeket). A kérdés ezek után csak az, hogy tudunk-e úgy választani bemeneti folyamatokat és tudunk-e úgy egymáshoz kapcsolni operátorokat, hogy a kiválasztott bemeneti folyamatok hatására az operátorok kimeneti folyamatai között legyenek olyanok is, amelyek számunkra hasznosak és hasznosítani is tudjuk ezeket. A válasz az, hogy tudunk. Az előbbi kérdésektől függ minden eszköz, minden gép, így a számológép hasznossága, illetve hasznosíthatósága is.

Műveletek gépi elvégzése

A matematikai műveletek elvégzése, azaz modellezése a számológép legfontosabb feladatai közé tartozik. A siker három dologtól függ. Attól, hogy hogyan modellezzük a kiindulási és eredményinformációkat, attól, hogy hogyan végezzük el (modellezzük) ezeket az információkkal való műveleteket, manipulációkat, végül pedig attól, hogy tudunk-e gondoskodni, és hogyan gondoskodunk arról, hogy az egymással kapcsolatban levő (valahogyan összetartozó) információk össze ne keveredjenek, a köztük levő kapcsolatok (számunkra) ismertek legyenek és maradjanak.

Vegyük szemügyre most operátorainkat! (Azaz először egyetlen pneumatikus operátorunkat és majd az ebből felépíthető – hozzá képest – összetett operátorokat.) Operátoraink végzik majd el az összes számítástechnikai műveletet, ami a számológépben (de bármely digitális berendezésben) előfordul. Érthető tehát, hogy legfontosabb feladatunk operátoraink működéstörvényének felderítése lesz.

Pneumatikus alapoperátorunk és matematikai modelljei

Pneumatikus alapoperátorunk *(1. ábra)* pontos működéstörvényét senki sem ismeri. Ennek ellenére rá vonatkozóan kiválóan használható és kívánt pontosságú – közelítő – működéstörvényeket, modelleket állapíthatunk meg. Ne felejtsük el, pontos működéstörvényét talán semminek sem ismerjük. Biztosan nem ismerjük például olyan eszközök pontos működéstörvényeit, mint a csavar, a csavarhúzó, a villáskulcs vagy a fogaskerék.

A legegyszerűbb működéstörvényt már említettük (Alapozás VIII.): „K akkor és csak akkor fúj, ha B_1, B_2, B_3, B_4 egyike sem fúj." Ez a működéstörvény nyilvánvalóan pontatlan. Az *„akkor és csak akkor"* azaz a *„pontosan akkor"* kapcsolat a matematikában gyakori, külön vadászunk is rájuk, mert nagyon hasznosak. Két jelenség egyidejűségét jelezzük így. Például „Egy háromszög akkor és csak akkor egyenlő oldalú, ha egyenlő szögű." Az egyenlőoldalúság és az egyenlőszögűség között azonban logikai, axiomatikai oksági kapcsolat van, nem pedig olyan fizikai kapcsolat, amelyben az egyik jelenség egy másikat fizikailag okoz, létrehoz.

Ha a kiváltó (fenntartó) jelenséget elég gyorsan követi a kiváltott (fenntartott) jelenség, és a fenntartó jelenség megszűnte után a fenntartott elég gyorsan megszűnik, akkor a két jelenséget egyidejűnek szokták venni. Ez a gyakorlat szempontjából nagyon sokszor nagyon előnyös becslés, nagyon hasznos közelítés, és ezért használni kell. Vannak azonban esetek, amikor az ilyen modell nemcsak, hogy pontatlan, hanem rossz is.

Alapoperátorunk – szintén használt – pontosabbnak hitt modellje a következő: „A bemenetek bármelyikén tapasztalt 'fúj' állapotot *bizonyos idő múlva* követi a kimeneten egy 'nem fúj' állapot." Ez a modell azonban rosszabb az előzőnél, mert egy hibát, pontatlanságot lehetetlenné tett ugyan, de más pontatlanságokat hozott helyette. E modell alapján nem hisszük már két jelenség pontos egyidejűségét. De mi az, hogy „bizonyos idő múlva"? És mi történik akkor, ha a bemeneteken megszűnik a „fúj" állapot? Visszaáll a kimeneten is a „fúj" állapot? Ha igen, mennyi idő múlva? Ez is „bizonyos idő múlva" történik? Ez a „bizonyos" és az előző „bizonyos" egyforma? És még sok más jogos kérdés is felvetődik.

Célszerű, ha matematikai eszközökkel tisztázzuk a kérdéseket. Tájékoztasson a szóban forgó rendszerjellemzők állapotairól egy állapotindikátor függvény, melynek két értéke lehet. A két értéket, a „fúj", „nem fúj" (a „huzat van", „nincs huzat") értéket 1, illetve 0 jelölje! A modell ekkor egy táblázattal megadható. (A *2. ábrán* csak 2 bemenet esetére adjuk meg a táblázatokat, először szavakkal, majd pedig számokkal.)

A „tökéletesített" modell a *3. ábrán* látható táblázattal egyenértékű. (Használják a t helyett az n-et is, és n-edik és n+1-edik állapotról beszélnek.) Az ábrán a t a t időpontot (időszakot?), t+1 pedig a következő (!) időpontot (időszakot?) jelöli, azt az időpontot, amit „bizonyos idő múlva" mutat az óra.

E modell értéktelenségét felesleges hangsúlyoznunk. Nagyon fontos azonban tisztán látni, hogy ennek nem a pontatlanság, hanem a definiálatlanság az oka. A modell nemcsak elméletileg, hanem gyakorlatilag is hibás. Ugyanis a kimeneti 1 vagy 0 állapot nemcsak a „bizonyos idővel" azelőtti állapot 0 vagy 1 voltától, hanem annak időtartamától is függhet. *És van, amikor függ is.* És olyan is van, amikor nem függ. Ezekben az esetekben általában akármilyen rövid lehet a kiváltó állapot fennállásának ideje, az állapotváltásnak azonban gyorsnak kell lennie, nem lehet az átmenet görbéje akármilyen lapos. Ilyenkor tulajdonképpen az állapotváltás a kiváltó jelenség. Mivel e kérdések konkrét megvalósításokhoz kötődnek, csak azok kapcsán tisztázhatók kielégítően.

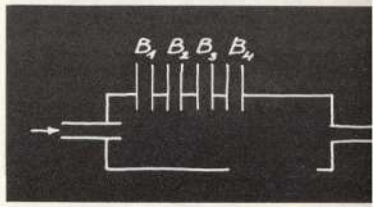
Célszerű, ha néhány egyszerű példával teszszük szemléletessé az előbbieket. Azt állítottuk, hogy fontos lehet egy jelenség időtartama is. Vegyünk egy folyót és képzeljük el, hogy folyásirányra merőlegesen hirtelen kivágunk belőle egy kilométer széles szeletet, és eltávolítjuk a szeletben levő vízmennyiséget a mederből. Ha a folyó folyása különféle módosításával közlünk információkat a folyásirány szerint alattunk álló megfigyelővel, ezt a jelenséget, illetve következményeit (hacsak nem túl messze van tőlünk) észre fogja venni. De ha tőlünk 1 méterre van is, de csak egy ezredmilliméteres szeletet vágnánk ki és távolítanánk el a folyóból, ezt már nem fogja észrevenni, pedig a folyó folyása, éppúgy, mint az előbb, megszakadt.

Nézzünk egy gépkocsisort egy egysávos úton! A közepéről több gépkocsi kanyarodjon ki egy mellékútra! A kocsisor foghíjassá válik. Igen, de meddig marad foghíjas? Nem sokáig, mert a foghíj utáni szakasz – saját törvényei szerint – felzárkózik. Hasonló a helyzet, ha egy középső kocsi rövid időre megáll, és feltorlasztja a kocsisor őt követő részét.

Természetesen a levegősugárból is kivághatunk, kilökhetünk részeket. És levegőáramlásra is igazak a felsoroltakhoz hasonló kiegyenlítődési, hézageltűnési jelenségek.

Mi ezek után a teendő? Alkossuk meg a pneumatikus alapoperátor pontosabb modelljét? Vagy a „tökéletesített" modellt tökéletesítsük tovább úgy, hogy a következő működéstörvényt mondjuk ki: *A bemenetek bármelyikén bizonyos ideig tartó „fúj" állapotot bizonyos idő múlva biztosan követi a kimeneten egy „nem fúj" állapot* (a „fúj" állapot kezdetét a „nem fúj" állapot kezdete).

Első pillanatra ez minden, csak nem matematikai modell. Pedig az. Természetesen nem a pneumatikus alapoperátorunk teljes körű működéstörvénye, hanem pneumatikus alapoperátorunk működéstörvénye, *de csak azokban az esetekben, amelyekben használják*. A többi esetre nyilvánvalóan nem igaz a modell, de ott nem is használják. És ezért nincs belőle baj. Az természetesen más kérdés, hogy illene megmondani, hogy ez nem a rendszernek a modellje, nem a rendszernek a viselkedéstörvénye, hanem csak modell, viselkedéstörvény *„bizonyos" esetekre*. Olyan ez, mint a feltételektől megfosztott kijelentés hangoztatása. („Nyugodtan törölje orrát a függönybe!" (Ejha!) Senki se botránkozzon meg! Csak el van hagyva a felszólítás elől egy olyasféle sor, mint let függöny $=$ „zsebkendő".)



1. ábra

Arra már figyelmeztettünk, hogy más a mérnöki és más a matematikusi gondolkodás. Ki-

csit másra és kicsit másképp használja a mérnök a matematikát, mint a matematikus.

A számológép-tervezőnek gondja van rá, hogy minden jelenség legalább „bizonyos" – és tudja, hogy az mennyi – ideig fennálljon a bemenete(ke)n, és azt is tudja, hogy legalább és legfeljebb mennyi a másik „bizonyos" idő, aminek elteltével biztosan a kívánt állapot lesz a kimenete(ke)n.

Természetesen járható az az út is, hogy a rendszernek egy minden esetben jól-rosszul használható modelljét, működéstörvényét is megalkotják. Igen, de minek. A bizonytalan helyzeteket, állapotokat a mérnök messze elkerüli, gondoskodik azokról a körülményekről, amelyek a berendezés rendeltetésszerű, megbízható működését biztosítják. A megbízható működés határán túli esetek kerülendők, és ezzel készen is van az egész.

Közismert, hogy az elektronikai berendezések működéstartománya minusz néhány foktól plusz valahány fokig terjed. A katonai berendezések működéstartománya valamivel bővebb. A megengedett hőmérséklet felett a gép elemei, és így a gép is, másképp kezd működni. Vagy ha belecsap a ménkű az épületbe, akkor is. Hogyan működik ilyenkor egy mikroprocesszor? Senki sem tudja. Azt sem tudja senki, hogyan és meddig működik a benzinmotor, ha a benzinbe kevés sósavat keverünk.

Azt tudjuk, hogy nem mindegy a sósav mennyisége. Az sem mindegy, hogy kijelentésünk érvényességi körét közöljük-e vagy sem. Azaz, hogy elmeműködésünk nyersanyagába pontos fogalmak és kapcsolatok közé mennyi definiálatlant vagy rosszul definiáltat (ez nem azonos a pontatlannal, amelynek pontatlanságáról *pontos* információinak vannak) hagyunk keveredni.

Nem csodálkoznánk, ha a matematikus alkotóakat tovább foglalkoztatnák egyes – a számológép szempontjából teljesen érdektelen és értéktelen – kérdések, az alapoperátor és társainak működésével kapcsolatban. Sokunkban él az igény, hogy milyen is lenne egy teljes körű működésmodell. Mivel ezt az igényt jogosnak érezzük, foglalkozni kell vele.

Mindenekelőtt tudatosítanunk kell, hogy igazán teljes körű modell megalkotására nem vagyunk képesek. Fizikai ismerethiány miatt sem. Senki nem próbálta ki a pneumatikus operátort olyan nagy nyomáson, amelyen például szétreped vagy deformálódásig hevül. Elektronikában sem próbálgatja senki nehezen megszerzett mikroprocesszorát az előírtnál lényegesen magasabb tápfeszültséggel járatni, hogy megtudja, mit bír ki. Hogy ezt megtudja, a berendezést odáig kellene feszítenie, amit már nem bír ki. De ez sem vinné előbbre, mert egy másik eszköz pontos tönkremeneteli határa biztosan nem lesz ugyanaz, mint az imént tönkretetté.

A tönkremenetel közelében általában „bizonytalanná" válik a rendszer. Azt sem tudjuk megmondani, hogy mit miért csinál. Ennek oka az, hogy a stabil működéstartományban sok rendszerjellemző hatása elhanyagolható, azon kívül azonban kezdik hallatni hangjukat.

2. ábra

B_1	B_2	K
nem fúj	nem fúj	fúj
nem fúj	fúj	nem fúj
fúj	nem fúj	nem fúj
fúj	fúj	nem fúj

B_1	B_2	K
0	0	1
0	1	0
1	0	0
1	1	0

3. ábra

$b_1(t)$	$b_2(t)$	$k(t+1)$
nem fúj	nem fúj	fúj
nem fúj	fúj	nem fúj
fúj	nem fúj	nem fúj
fúj	fúj	nem fúj

$b_1(t)$	$b_2(t)$	$k(t+1)$
0	0	1
0	1	0
1	0	0
1	1	0

A rendszer látszólag sokkal több változós kezd lenni. Ez azonban nem igaz. Mindig is sokváltozós volt, csak most már nem hanyagolhatjuk el azok többségét. A nagyon sok változós rendszereket pedig alig-alig vagy sehogyan sem tudjuk áttekinteni. Áttekintés nélkül viszont reménytelen a modellalkotás. Ezek fontos tények, és nem tehetetlenségünk megmagyarázására szolgáló ideológia.

Mivel mégsem vagyunk teljesen tehetetlenek, tovább foglalkozunk pneumatikus operátorunk modelljeivel, addig, amíg megnyugtató működéstörvényt nem sikerül előállítanunk.

Pontosabb modell, pontosabb működéstörvény

Lássuk először az első modellt! Használjuk a matematika eszközeit a kapcsolatok leírására! Most is csak 2 bemeneti folyamatot veszünk figyelembe a 4 helyett. Ez azonban nem korlátozza sem mondanivalónk lényegét, sem általánosságát. A bemeneti folyamatok legyenek $b_1(t)$, $b_2(t)$, a kimeneti pedig $k(t)$.

A rendszer modellje, azaz a pneumatikus operátor működéstörvénye pontosan az

$1-\max(b_1(t), b_2(t))=k(t)$

összefüggés. Ugyanezt másképp,

$1-(b_1(t)+b_2(t)+\text{abs}(b_1(t)-b_2(t)))/2=k(t)$

formában is kifejezhetjük. Sőt a

$\min(1-b_1(t), 1-b_2(t))=k(t)$

összefüggés is ugyanazt fejezi ki, és ezt is lehet több más, például

$(2-b_1(t)-b_2(t)-\text{abs}(b_1(t)-b_2(t)))/2=k(t)$

alakban is megadni.

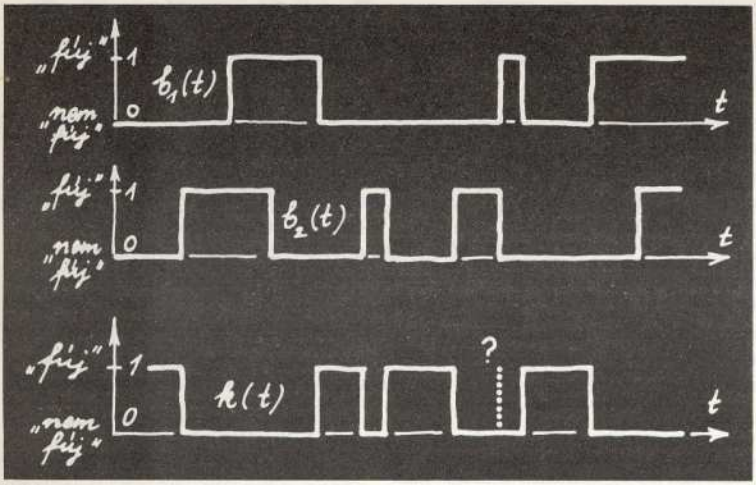
Egy-egy konkrét folyamatot, viselkedéstörténetet szemléltet a *4. ábra*. Az ábra folyamatai természetesen mind a négy egyenlőséget kielégítik. Ezzel elintézettnek is szokták tekinteni a problémát, pedig nem egészen az. A matematikai formula egyértelműen rendelkezik ugyan minden esetre, de a működéstörténet vonalainak t tengelyre merőleges szakaszai gyakorlati értelmezési problémákat vetnek fel.

Az *5. ábra* matematikailag megengedett lehetőségei közül például melyik áll fenn a valóságban? Nem tudjuk. Azt sem tudjuk, hogy a kérdés értelmes-e (értelmezhető-e) egyáltalán.

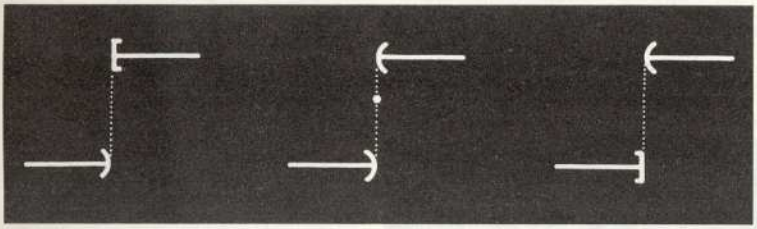
Mi eljutottunk a kérdésfelvetésig. Válaszolni rá nem tudunk. De ez a gyakorlat szempontjából szerencsére nem is mindig fontos, mert a bizonytalan lezajlású állapotváltozások közvetlen közelében nem szoktunk mintát venni, mint ahogyan nem szoktuk a dobókockát röptében vagy mutatót lengés közben leolvasni, mert ezek eredménye bizonytalan lehet.

Rátérve második, „pontosabb" modellünk matematikai formába öntésére, ez érdekes, és e fejtegetések körét lényegesen meghaladó jelentőségű tanulságokhoz fog vezetni, melyeknek lényege az, hogy nem tudunk (helyesebben csak nehezen tudnánk) a szavakban megadott modellel egyenértékű, keveset mondó matematikai modellt megadni. A legegyszerűbb matematikai modell is többet mond, mint a szavakkal megfogalmazott.

E modellt, a „tökéletesített" működéstörvényt az $1-\max(b_1(t), b_2(t)) \Rightarrow k(t+a)$ összefüggés adja meg. Az a az a „bizonyos idő", a „késés", amelynek elteltével a kimeneti jelenség a bemenetit követi. Nagyon fontos szerepe van az összefüggésben az irányított egyenlőségnek, amely azt fejezi ki, hogy a baloldal határozza meg a jobboldalt. Ha a baloldal értelmes, akkor a jobboldalt meghatározza. Abból azonban, hogy a jobboldal értelmes, a baloldalra



4. ábra



5. ábra

semmiféle következtetést levonni nem szabad. Célszerű ezt a megkülönböztetést a matematikában is használni, például

$$\log(a)+\log(b) \Rightarrow \log(ab).$$

A tapasztalat azt mutatja, hogy a formulánk „jobbról balra" is igaz, azaz nagy pontossággal igaz az, hogy

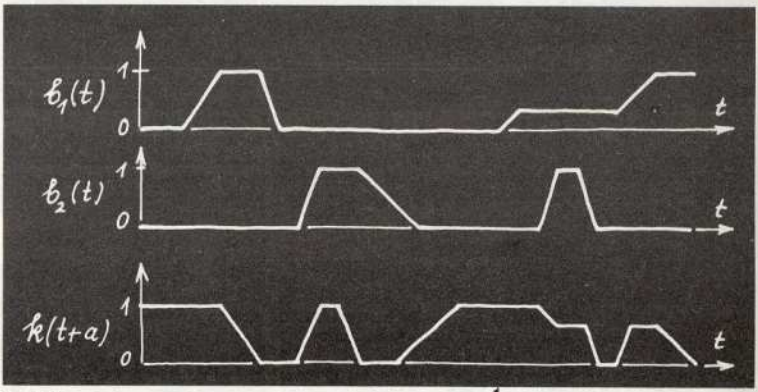
$$k(t+a) \Rightarrow 1-\max(b_1(t), b_2(t)).$$

Tehát

$$1-\max(b_1(t), b_2(t)) = k(t+a)$$

is fennáll „nagy pontossággal". Ez az összefüggés azonban többet mond, mint a szavakban megfogalmazott működéstörvény.

Először is egyetlen „bizonyos idő" van. Azaz az 1-ből 0-ba és a 0-ból 1-be való átmenetet követő következmények egyforma idő múlva jelentkeznek. Másodszor nemcsak az állapotváltozások közötti kapcsolatról, hanem az állapotok fennállására vonatkozóan is világosan és közvetlenül informál. Egy-egy állapot fennállására a szóbeli modellből külön kellett következtetni. Végül pedig az állapotváltozás lebonyolódására vonatkozóan is kapunk információt. Az, hogy egy-egy csőben „huzat van" vagy „nincs huzat" és hogy most éppen egy cső „huzatosból" „huzatnélkülivé" vagy fordítva alakul, szintén fontos lehet. Az állapotváltozások



6. ábra

módjára vonatkozó, a modell által szolgáltatott információkat szemlélteti a *6. ábra* egy konkrét működéstörténet esetében.

A bemeneti állapotváltozások módja meghatározza $k(t+a)$ állapotváltozásának módját.

Az állapotváltozásokra vonatkozóan a modell azonban nagyon pontatlan. De mivel az állapotváltozások környékét (most!) kerüljük, ez nem okoz hibát.

A modell pontatlan a túl gyakori és túl meredek állapotváltozások esetében is. Ez oly mértékű is lehet, hogy használhatatlannak kell nyilvánítanunk.

Noha képesek lennénk pneumatikus alapoperátorunk működését ilyen esetekben is használható modellel leírni, erre azért nincs szükség, mert az ilyen eseteket a számológép-szerkesztő a már említett „összemosódási", „eltűnési" jelenségek miatt elkerüli.

Ezzel eljutottunk addig, hogy pneumatikus alapoperátorunk viselkedését matematikai formulából is meg tudjuk határozni, nem kell mindent megépíteni és kísérletileg eldönteni. Hozzáfoghatunk majd digitális berendezést tervezni íróasztal mellett. Biztosak lehetünk abban, hogy ami matematikailag „rendben van", az fizikai valóságában is elképzelésünknek megfelelően fog működni.

Fő eredményünk az, hogy sikerült kapcsolatokat megállapítani a rendszerjellemzők, illetve ezek változási folyamatai között. Ezek a kapcsolatok matematikai függvénykapcsolatok voltak, azaz olyan kapcsolatok, amelyekben a változóknak csak egyetlen értéke szerepelt.

A természetben más kapcsolatok is vannak, amelyek nem írhatók le úgy, hogy változónként egy-egy érték van kapcsolatban egymással, hanem a jellemzők alakulása, tehát előző értékei is szerepet kapnak. A legegyszerűbb és leggyakoribb ilyen kapcsolatban egy-egy változó változási sebessége – regisztrátumgörbéjének pillanatnyi meredeksége – van matematikai függvénykapcsolatban más változók értékével.

Ezekben az általánosabb esetekben például nem egy-egy érték határoz meg egy másikat, hanem folyamatok egészében határoznak meg egy értéket vagy egy másik folyamatot. Az ilyen kapcsolat nem adható meg egy „helyettesítési" értékek közötti matematikai függvénnyel. Ezekről később még szó lesz.

Miután belekóstoltunk a működéstörvénykészítésbe, a legegyszerűbb esetek egyikében, és sikerült megalkotnunk egy gyakorlati céljainknak megfelelő modellt, munkánkat a konkrét gépépítés első lépéseivel folytathatjuk.

POGÁNY CSABA

Racionálisabbá teheti az ellenőrzést és a munkaidő kihasználását, ha

TZ–80 SZÁMÍTÓGÉPPEL VEZÉRELT MUNKAIDŐ-RÖGZÍTŐ ÉS NYILVÁNTARTÓ RENDSZERT

alkalmaz vállalatánál

Felhasználási terület:

– munkaidő-nyilvántartás
– beléptető és követő rendszer
– gépjármű-telephely irányítás stb.

A rendszer különleges szolgáltatásai:

- Az „események" rögzítése a számítógépet nem foglalja le, a „háttérben" történő munkaidő-nyilvántartás mellett a számítógép egyéb feldolgozásra is felhasználható
- Rugalmas bővíthetőség, mind a terminálok számában, mind a programok tekintetében
- A feldolgozott adatok segítségével egy újabb programcsomag elvégzi a személyi nyilvántartást és a bérszámfejtést is

A rendszerrel kapcsolatos részletes tájékoztatással Kereskedelmi Főosztályunk az érdeklődők rendelkezésére áll!

INFORMÁCIÓTECHNIKAI VÁLLALAT
Kereskedelmi Főosztály
Budapest III., Kerék u. 6.
T.: 803-294

Mit tehet a már nem használható bizalmas írásokkal és dokumentációkkal?

Meg kell semmisíteni!

Ehhez ajánljuk az AKTEMA–V típusú iratmegsemmisítőt

– Az A/4 formátumú iratokat gyorsan és megbízhatóan olvashatatlanná teszi
– A papírörlemény értékes másodlagos nyersanyag marad a népgazdaságnak
– Optimális teljesítmény, 40 000 A/4-es lap óránként
– Minden irodai munkahelyen felhasználható

Részletes tájékoztatást ad:

INFORMÁCIÓTECHNIKAI VÁLLALAT

Kereskedelmi Főosztály
Budapest III., Kerék u. 6.
T.: 803-294

HAZAI MIKROPIAC '84

Az Orgtechnik kiállításon egy kényelmes fotelekkel körülvett dohányzóasztalon aranyszínű minifloppy szobor állt. Ezt az Aranyfloppy-díjat szántuk 1984 legjobb, legnépszerűbb hazai gyártású mikroszámítógépének.

Alábbi áttekintésünket választ szánjuk arra a kérdésre, hogy milyen volt Magyarországon a mikro-számítástechnikai piaci élet 1984-ben. Elöljáróban, az elméleti fejtegetéseket kerülve, csupán a közgazdasági tartalomra emlékeztetve megjegyezzük, hogy élénk piac ott van, ahol sokféle, jó minőségű termék versenyez egymással.

Esetünkben a sokféle igaz, de a jó minőségű termék kritériumai még gyakran hiányoztak. Vizsgáljuk külön a hardver- és a szoftverpiacot!

Hardvertükör

A hobbi kategóriában megjelent az első hazai mikrogép, a Primo. Sajnos nem az igért 3 ezer darab, hanem ennek csak alig a fele került az év végéig a felhasználókhoz. A gép ára a kiépítéstől és a vevő kilététől (közület, magánszemély) függően 12–25 ezer forint. A Sinclair ZX81-et összel a Skála–Metróban 12 ezer forintért, Bécsben 999 schillingért lehetett kapni. A megbízhatóság mindkét géptípus esetében csak a hobbi igényeknek felel meg.

A félprofesszionális kategóriában a Sinclair Spectrum, a VC–20, a Commodore 64, a hazai gyártásúak közül az Aircomp–16, a Mickey 80 és a HT–1080Z volt a meghatározó a piaci forgalomban. Az árak 30 és 50 ezer forint között szórnak. (A µMagazin szerkesztőségében is sok vita van arról, hogyan is lehetne a hazai piacon kapható személyi számítógépeket kategorizálni. A megbízhatatlan hajlékonylemezzel ellátott, hazai gyártású „professzionális" gép aligha professzionális. A néhány pontján – mint a képernyő, a „sokmadzagos" összeépítés – nem professzionális Commodore 64-et viszont még olyan helyen is professzionális gépként tudták használni, ahol az előbbi gépet aligha lehetett volna. A hiány és kiszolgáltatottság azonban olyan helyzetet szült, hogy a professzionális/vállalati szféra a Nyugaton csak házi alkalmazásokra készült gépeket, például a VC–20-at és a ZX–Spectrumot is kénytelen volt megvásárolni. Ennek a meglehetősen visszás helyzetnek a visszatükröződését látjuk a cikk szerzőjének szerintünk realista kategorizálásában és a magunk részéről ilyen értelmezésben tudjuk ezt elfogadni. – A szerk.)

A félprofesszionális kategóriában mintegy 15 ezer gazdálkodó egység, iskola és magánszemély vált géptulajdonossá 1984-ben. Az említett típusok közül a Commodore 64 lett a legelterjedtebb személyi számítógép. Nem véletlenül: könnyen beszerezhető, szolgáltatásokkal jól ellátott, kedvező bérletvételi (leasing) feltételekkel kínált gép. Az egyetlen probléma – de ez nem a piachoz tartozik –, hogy a számítástechnikával ujonnan ismerkedő szervezetek túlzott követelményeket támasztanak vele szemben.

A legnehezebb a professzionális mikrogépek piaci helyzetét jellemezni, mert itt szérianagyságról nemigen beszélhetünk. A hazai és a szocialista országok fejlesztés alatt álló gépeit, gyártási terveit viszont valóságnak hisszük, és így túlságosan színes képet kapunk. Ezért csak áttekintjük a tavalyi helyzetet.

A tőkés gépek közül szinte minden típus kapható megrendelésre, 2–3 hónapon belül, a bizományosi hálózaton keresztül. A legkeresettebb konfigurációk a Commodore 700-as és 8000-es sorozat, valamint az IBM PC. Egy Commodore 720-as nagy konfiguráció (alapgép képernyővel, 2 Mbájtos duál-hajlékonylemez, nyomtató) ára 800 ezer forint, az IBM PC/XT (képernyő, billentyűzet, 10 Mbájtos merevlemez, 360 kbájtos hajlékonylemez, nyomtató, BASIC, Assembler és DOS 2.0 alapszoftver) ára 2,1 millió forint volt. Kelendőek voltak a kuriózumok is: a BÁV Elektronikai Áruház 3 Macintosh konfigurációt értékesített.

A hazai gyártásból két olyan típust emelhetünk ki, amelyre érvényes volt a jó hozzáférhetőség és a hazai viszonylatban nagyobb darabszám: a Proper családot és a TAP 34-et. Az SZKI a Proper–8 számítógépből 100 darabot, a Proper–16-ból 160 darabot adott el 1984-ben. A Telefongyár TAP 34 számítógépéből 135 darabot vásároltak meg hazai megrendelők (ennek majd háromszorosát gyártották exportra). Jó kereskedelmi paraméterekkel jelent meg a TZ80, és újdonság volt a hordozható változat, a TRANSMIC.

Változatlanul nem volt számottevő a szocialista import: az IZOT 0220M2-vel és az SZM 1300-zal próbálkoztak.

1984-ben mindent egybevetve 1500–2000 professzionális mikrogépet adtak el.

Szoftvertükör

Felméréseink szerint 800–1000 professzionális mikrogépes szoftvertermék van forgalomban. A szoftverpiacra a kidolgozók és forgalmazók olyan méretű decentralizáltsága a jellemző, hogy a felhasználó nem is érzi a tényleges kínálatot. Segített ezen, hogy az elmúlt évben megjelent két mikroszoftver-katalógus – a KSH és az LSI ATSZ összeállításában –, amelyek üzletben kaphatók.

A fejlesztő intézetek közül az SZKI és a SZÁMALK OSAK, a forgalmazók közül a Novotrade kínál olyan katalógust, amelyből választani lehet. A Novotrade katalógus 120 programtermékket tartalmaz Commodore 64 gépre. Átlagáruk 15 ezer forint. A hobbi kategória szoftvertermékeire jellemző a 2000 forintos alapszoftver és a 300–500 forintos játékkazetta-ár. A szoftverpiac legkényesebb pontja a felhasználói érdekvédelem hiánya és a minőség rizikója.

Az értékesítés módjai

1984-ben a kiskereskedelmi hálózat kialakulása jelentős piacélénkítő tényező volt a mikro-számítástechnikában. Az év végére mintegy 20 szakbolt nyílt, ahol utcán át lehet mikrogépet, szoftvert, kiegészítőket vásárolni. A boltok a legkülönfélébb gazdálkodási formában (kisszövetkezet, bizományosi hálózat, fejlesztő intézeti társulások stb.) működnek. A szakboltok nagyobb része bizományosi tevékenységet is ellát, ezáltal teremtve meg saját kínálatát.

Az üzletek forgalmának megoszlása 90–10 százalék a hardver javára. Ezért említünk meg egy, a szoftver kiskereskedelem terén lebonyolított sikeres akciót. A Budapesti Skála Nagyáruház galériáján két hónapig tartott nyitva a Novotrade Rt. szoftverbemutató terme, ahol működés közben egyszerre 10–15 szoftvertermeket tekinthettek meg a vásárolni szándékozók, és vehettek is meg azonnal egységes kivitelben, jól dokumentáltan.

A segédanyag-ellátásban voltak problémák az év folyamán, de az év végére jelentősen javult a helyzet. Nagymértékben hozzájárult ehhez az ÉSZ (ÁPISZ–SZÁMALK) országos bolthálózat budapesti üzletének (XI., Budafoki út 7.) megnyitása, aminek hatására helyükre kerültek a hajlékonylemez-árak.

Egyelőre hiánypiac

Az 1984. évi mikropiac magán viselte a hiánypiac összes ismérvét. A mikro-számítástechnika piaci forgalma 1,5–2 milliárd forint volt, ami 60 százalékban tőkés eredetű, nagyrészt a bizományosi hálózaton keresztül értékesített termékek árából tevődött össze. A piac fejlődését elsősorban a spontaneitás jellemezte.

DR. ZÁRDA SAROLTA

Most elmeséljük, miért került a zsüri olyan helyzetbe, hogy végül is egyik gépnek sem ítélte oda a díjat.

A versenyt a következőképpen akartuk lebonyolítani. Az olvasók szavazatai alapján képet kapunk a felhasználói szempontok értékeléséről, egy szakértő bizottság pedig megállapítja az objektív, összehasonlítható műszaki paraméterek szerinti rangsort, és a kettő összevetéséből adódik az eredmény.

Az olvasói szavazatok azonban kis számuk és nem kellően reprezentatív voltuk miatt nem töltötték be a nekik szánt szerepet. Erre tulajdonképpen fel voltunk készülve, és mikroszámítógép segítségével próbáltunk eligazodni, a szempontokat súlyozni. Így kerültünk kapcsolatba dr. Ráduly Zoltánnal, a BME Ipari Üzemgazdaságtan Tanszék adjunktusával, aki felajánlotta a náluk kidolgozott és több helyen jól bevált rangsoroló eljárás alkalmazását az év mikroszámítógépének kiválasztásához. Az eljárás folyamata az alábbi volt:

1. Az értékelési tényezők feltárása csoportmódszerrel
2. Az értékelési tényezők súlyozása páros összemérés módszerével mikrogéppel
3. A termékek minősítése a mérhető adatok alapján
4. Egyéni zsürizés a nem mérhető szempontok esetében, a minősítés kiegészítése
5. Rangsorolás mikrogéppel

A rangsor megállapításában közreműködött egy 12 főből álló zsüri, a rangsoroló eljárás levezetői (Ráduly Zoltán és négy villamosmérnök-hallgató), valamint egy Commodore 64-es konfiguráció néhány képernyővel és 12 szavazóbillentyűvel.

A versenyen 20 hazai gyártású mikroszámítógép vett részt, három: hobbi, személyi és professzionális-moduláris kategóriában. Az első nagy feladat a kategorizálás volt, amit végül is a következőképpen oldottunk meg:

Hobbi gépek	Személyi számítógépek	Professzionális-moduláris számítógépek
Aircomp 16	HT–108Z	HT 680OX
Primo	Floppymat SP	Proper–16
	SLK 80	Comput 80
	Proper–8	MOD81
	MO8X	TRANSMIC
	TAP 34	TZ 80
	Syster	TPA Quadro
	Rosy 80F	Janus
	Labsys 80	
	VPPC	

AZ ÉV GÉPE (AZ ÉV KÍSÉRLETE)

Az értékelési tényezők feltárása mintegy négyórás közös munka eredménye volt. Ennek során 58 szempontot vizsgáltunk, melyeket 18 csoportba és két főcsoportba soroltunk. Ezután a szempontok a két főcsoporton belül „körmérkőzést" játszottak szavazataink és a számítógép segítségével, végül a főcsoportok arányát határoztuk meg egymással szemben.

Az értékelési rendszer kialakításánál a számítógép vizsgálta az egyes zsüritagok válaszainak ellentmondásmentességét (ez mindenkinél 80% felett volt), és kiszámította a zsüritagok egyetértési együtthatóját is, ami az alkalmazási és egyéb szempontok főcsoportban 97,54, a hardver/szoftver főcsoportban 98,85% volt.

Az értékelési tényezők és súlyszámaik az alábbi módon alakultak ki.

A szoftver- és hardverszempontok, csoportsúlyszámok

1. Szabványosság 4,93
   - A szabványoknak (CP/M, MS-DOS, egyéb nemzetközi) való megfelelés
   - A szoftver hordozhatósága
   - Gyártótól független szoftverellátás
2. Magyar ékezetes betűk kezelése 3,63
3. Rendszerszoftver-ellátottság 5,72
   - A BASIC utasítások száma
   - Assembler programozási lehetőség
   - Gépi kódú hordozhatóság
   - Nyelvi rendszerek
   - Direkt fájlkezelés lehetősége
   - A fejlesztést támogató szoftver ellátottság
   - Alkalmazási célszoftverek
4. MEI minősítés 2,97
5. Háttértárak 5,96
   - Típus, kapacitás
   - Alapkiépítésben
   - Egyidőben hányat kezel
6. Az alkalmazott processzor 3,37
   - Tárkapacitás, bővítés
   - A központi egység sebessége
   - A mikroprocesszor típusa
7. Képernyő 4,04
   - Felbontás (karakteres, grafikus)
   - Színek száma
   - Minőség (vibrálás, tükröződés)
8. Billentyűzet 4,22
9. Konfigurálhatóság 5,2
   - Bővíthetőség
   - Interfészek
   - Új perifériák illeszthetősége
10. Üzembiztonság 6,1
    - Megbízhatóság (MTBF, folyamatos üzemóra)
    - Javíthatóság (Ft, gyorsaság, MTTR)
    - A berendezés integráltsági foka (elektronika)

Az alkalmazási és egyéb szempontok, csoportsúlyszámok

1. Szerviz 7,79
   - Szervizellátottság
   - Vevőszolgálat
   - Cseregép-lehetőség
2. Alkalmazóiszoftver-ellátás 6,85
   - Standard szoftvercsomagok száma
3. Hálózati alkalmazhatóság 5,4
   - TAF illeszthetőség
   - LAN illeszthetőség
4. Felhasználó-orientáltság 8,14
   - Adatbiztonság
   - Bolondbiztonság
   - Segédanyag-szükséglet és ellátás
   - Elégedettségi szint
   - BASIC didaktikusság
   - A hibajelzések egyértelműsége
5. Ár 6,52
6. Kompatibilitás 7,52
   - Más gépekkel
   - Korábbi típusokkal
7. Formakultúra 4,3
   - Ergonómia
   - Súly, hordozhatóság
8. Értékesítési jellemzők 6,74
   - Az eddig összesen eladott mennyiség
   - Az 1984-ben eladott mennyiség
   - Szállítási határidő
   - A sorozatgyártás bejelentésének éve
   - Dokumentálthatóság
   - Gépspecifikus oktatás, hozzáférhetőség

Ezt a szempontrendszert bizonyára minden leendő felhasználó hasznosan alkalmazhatja gépkiválasztásnál. Érdekes megfigyelni, hogy a két legnagyobb súlyszámot a felhasználó-orientáltság és a szerviz kapta – ezt viszont a fejlesztők figyelmébe ajánljuk.

Az értékelésben ezután következett az adatok begyűjtése a gyártóktól. Az adatgyűjtést a valóban szakembereknek számító egyetemi hallgatók végezték, akik a műszaki részletkérdésekben is otthon vannak. Az adatok hitelességét minden esetben aláírás, pecsét igazolta. Mégis meg kell jegyezni, hogy ennek ellenére az értékesítési darabszámot és a javítási időt tekintve „csodákat" hallhattunk.

A nem mérhető szempontokról ismét a zsüri döntött. Ilyen volt például a formakultúra, a dokumentáltság szintje stb. A beérkezett adatok alapján két gépet ki kellett zárnunk a versenyből: a VPPC-t megfelelő információszolgáltatás hiányában nem tudtuk értékelni, a Janus eladott darabszáma pedig nem érte el az 50-et, így nem tekinthettük az 1984. évi kínálat összetevőjének.

Már csak a végeredmény van hátra. Az élenjáró gépek kategóriánként: a Primo, a Proper–8, a TPA Quadro. És hogy korrigáljuk az esetleg vitatható kategorizálást, felsoroljuk az összevont értékelés első tíz helyezettjét:

Géptípus	Rang
TPA Quadro	59
Proper–8	58
Proper–16	55
TAP 34	53
SLK 80	52
MO8X	46
TRANSMIC	46
Primo	45
Labsys	44
Floppymat SP	44

A zsüri és a szerkesztőség ezek után úgy döntött, hogy nem adja ki az év számítógépe díjat. A döntést a következők indokolják.

1. Az olvasói szavazatok kis száma miatt külön reprezentatív felmérést kellett volna végezni a felhasználóknál. Ez viszont meghaladta a vállalkozás eredetileg tervezett kereteit.

2. A gyártóktól nem sikerült kellően egzakt és összevethető adatokat beszerezni termékeikkel kapcsolatban, annak ellenére, hogy az adatgyűjtést végzők lelkiismeretesen és mindenre kiterjedően végezték munkájukat.

3. A rangsoroló eljárás csak egy döntést segítő és nem döntést hozó eszköz. A termékekkel kapcsolatos ismert műszaki, kereskedelmi, vevőszolgálati, szoftverellátási és alkalmazhatósági adatok viszont nem voltak elég pontosak és összemérhetők ahhoz, hogy a rangsoroló eljárás kellően megalapozott segítséget adott volna a végső döntéshez.

4. Sem a zsüri, sem a szerkesztőség nem érezte ezek után magát kellően felkészültnek arra, hogy a szubjektív megítélést és saját tájékozottságát segítségül híva hozzon döntést.

Vállalkozásunkat kísérletnek tekintettük. Célunk az volt, hogy megtegyük az első lépést a hazánkban egyre nagyobb számban gyártott és forgalmazott mikroszámítógépek reális értékítéletének kialakítása felé. Sajnálattal kellett azonban megállapítanunk, hogy a hazai mikroszámítógépek piacát még mindig az esetlegesség jellemzi. Mind a gyártók, mind a felhasználók részéről hiányzik még az igazi vevő–eladó viszony, így szinte törvényszerű, hogy a sokrétű, igazi piaci értékítélet csak most van kialakulóban.

Reméljük, hogy kísérletünk legalább azzal az eredménnyel járt, hogy valamelyest felszínre kerültek azok a problémák, amelyek akadályozzák egy reális hazai értékítélet kialakulását. Csak ezeket a problémákat megoldva tudjuk jövőre folytatni mostani kezdeményezésünket.

Személyi számítógépek

Több mint 20 millió személyi számítógép került alkalmazásba az elmúlt évek során. A gyártók eddigi stratégiájuk kialakításánál elsősorban az egyént vették célba. Ennek megfelelően alakultak a piaci viszonyok és a személyi számítógépek kategóriái is.

Tarol az IBM gőzhengere

1983-ban még hiánycikk lehetett az IBM személyi számítógép. Akkoriban volt olyan időszak is, amikor csak „jattért" lehetett azonnal géphez jutni az ilyesmihez egyáltalán nem szokott Egyesült Államokban.

Az IBM ragyogó piaci propagandát fejtett ki. Chaplin csavargójának figurájával és az asztalon elhelyezett piros rózsával emberközelivé, sőt az egyén számára vonzóvá tette az IBM PC-t. A reklám eszébe juttatta mindenkinek, hogy a Modern idők című filmben még elesett kiseminberként viselkedő csavargó a legújabb idők technikáját teljesen otthoniasnak érzi, és nincs vele szemben semmi emberi fenntartása.

Az IBM a független szoftvergyártókat már korábban maga mellé állította gépének nyílt architektúrájával és maximális együttműködési készségével. Meg is jelent 1983-ban a személyi számítógépes szoftverek olyan áradata, amilyet még nem látott a mikrovilág.

A már korábban csatasorba állított független kiskereskedelmi hálózat (retail chains, dealers) egymás után adta fel az egyre nagyobb mennyiségekre szóló megrendeléseket. A 10 Mbájtos Winchesterlemezzel ellátott PC XT változat megjelenése még csak fokozta az étvágyat, különösképpen az üzleti és professzionális alkalmazások területén. A hosszas „suttogó" propagandával előkészített kisöcs, a PCjr év végi bejelentése az igényesebb otthoni és az oktatási alkalmazások területén kívánta az IBM számára kikaparni a gesztenyét.

Ilyen előzmények után 1984 még inkább az IBM éve volt, mint a megelőző. Az 1983-ban eladott 675 ezer személyi számítógéppel szemben 1984-ben majdnem 1,7 millióra becsülhető az IBM személyiszámítógép-eladások száma. Ráadásul 1983-ban még mindössze 45 ezer PC XT modellt adtak el, míg 1984-ben ennek legalább a tizszeresét.

A felhasználóknál levő IBM PC-knek még ennél is nagyobb hányada nyert azonban nagy kapacitású Winchester háttértárat, hiszen a nyílt architektúra lehetőségeit a független perifériagyártók is kihasználták.

A jelszó: IBM kompatibilitás

A személyiszámítógép-gyártásba bekapcsolódó új cégek sorozatban döntöttek úgy, hogy IBM PC kompatibilis gépet fognak gyártani. A Compaq, a Corona, a Columbia, az Eagle csak a legjelentősebbek az újonnan alapított vállalatok közül. A Televideo a displayterminálok gyártásában ismert egyik vezető cég. A hagyományos számítógépgyártók közül a Sperryt, a telekommunikációból a személyi számítógép-gyártásba is bekapcsolódó gyártók közül az ITT-t kell megemlítenünk.

Külön kell szólni a maga Unix kompatibilis világát komoly reklámköltséggel forszírozó AT&T-ről. A helyi telefontársaságok 1984. januári leválasztása után számítógépgyártásra is jogosulttá vált cég a Unix System V-öt használó asztali és miniszámítógépek mellett az IBM PC kompatibilis 6300-as modellel jelent meg a piacon.

Ez a gép az időközben részben felvásárolt Olivetti cég M24 modellje alapján készült. Külön előnye, hogy a memória sínrendszerben 16 bites adatút-szélességet használ, 8 MHz-es 8086 processzoron alapszik, és a képernyő grafikus felbontása is duplája az IBM PC-énck (640 × 400 16 színben). A gép teljesítménye így szinte minden jellemzőjében két-háromszorosa az IBM PC-énck. Bővítési sínrendszere ugyanakkor kompatibilis az IBM PC-vel is, és szoftver tekintetben is képes azokat a speciális programokat futtatni, amelyek a kompatibilitás alapvető ismérvei.

Képünkön az Olivetti M24 típusú személyi számítógépe látható, az Alloy Computer Products nagy teljesítményű háttértár-rendszerével kiegészítve. A PC–Stor nevű kiegészítő rendszer további 20–85 Mbájt Winchester-kapacitást és mágnesszalagos gyors mentő/töltő alrendszert ad, kazettánként 21,5 Mbájt kapacitásban.

Mindez igen jól mutatja azt a piaci taktikát, hogy az IBM PC kompatibilis gyártók egy része a nagyobb teljesítőképességgel és kiépíthetőséggel próbálja javítani piaci pozícióit. Ez nem véletlen, mivel az utóbbi időben egyre több bírálat éri az IBM PC teljesítőképességét, és nem is jogtalanul.

Apple – IBM párharc

Az Apple cég nemcsak úttörője volt a személyi számítógépes technikának, hanem egész 1981-ig képes is volt megőrizni vezető helyét. Az IBM PC megjelenésekor nem látta különösebben veszélyeztetve pozícióját, mivel technikai értelemben valóban helyesen mérte fel, hogy az Apple II-höz képest nincs semmi minőségi újdonság a PC-konstrukcióban. A 16 biteskent reklámozott IBM gép is csak 8 bites adatsínt használ (csak a processzoron belüli adatutak 16 bitesek, így 8/16 bites gépnek kellene nevezni), és az Apple II grafikai lehetőségei sem rosszabbak, legalábbis a korszerűsített változatban.

Az Apple két irányban látta a továbblépést. Egyrészt dolgozott az Apple III konstrukción, másrészt a valóban új minőség elérését célul kitűző Lisa-n (Local Integrated Software Architecture). Az Apple III a 8 bites technológia elbonyolított továbbfejlesztésének bizonyult, így szükségszerűen buknia kellett.

Az Apple-t az Apple II jól sikerült továbbfejlesztése mentette meg. Az Apple IIe modell 31 integrált áramkört tartalmazó alapkártyája a korábbi Apple II három kártyán elhelyezett 109 integrált áramkörét váltotta fel. Így az Apple II vonal 1983-ban is sikerrel folytathatta hódító útját. Az ok: kialakult szoftverbázis és vevőkör, jelentősen csökkentett előállítási ár, valamelyest növelt teljesítmény. Az eredmény: 750 ezer darabos eladás 1983-ban.

Annál nagyobb csalódást okozott a Lisa. Az igen jelentős propaganda és a valóban lelkes szakmai fogadtatás ellenére alig 20 ezer Lisa-t sikerült értékesíteni 1983-ban. Hiába kapott a felhasználó a géppel együtt hét olyan általános alkalmazási programot, amelyekkel hagyományos programozás nélkül adaptálni tudta a gépet az üzleti/professzionális szféra feladataira. Hiába működtek az alkalmazások egymással egyeztetett, integrált módon. Hiába könnyítették a gép kezelését valóban forradalmi módon (közvetlen manipuláció, vizuális kapcsolat stb.).

Vannak, akik az igen magas, 10 ezer dolláros induló árat, vannak, akik az ilyen eszközrendszer iránti igazi felhasználói igények hiányát, mások a túl „csicsásnak" és mesterkéltnek bizonyult Lisa-környezetet tartották a kudarc okának.

Az Apple a Lisa-technológia végletes leegyszerűsítésében vélte meglelni a kiutat. A 128 kbájt RAM és 400 kbájt hajlékonylemez alapkapacitású Macintosh gépbe igyekezett sikerrel bepasszírozni a Lisa alapvető sajátosságait. A független szoftvergyártókkal még a gép piaci bevezetése előtt felvette a kapcsolatot, és folyamatosan buzdította őket az együttműködésre. Az ily módon minimalizált kiépíthetőség mellett maximalizált teljesítőképességű gépet egyúttal olyanra is tervezte, hogy az előállítása is a lehető legkevesebbe kerüljön. Állítólagos 350 dolláros önköltségi árával a Macintosh így alighanem a legolcsóbban előállított és egyúttal a legkorszerűbb személyi számítógép az üzleti/professzionális szférában.

A Mac-nek is becézett, jópofa gépből 280 ezer darabot adtak el 1984-ben. Ez ugyan elmarad az eredetileg tervezett 400 ezer darabos álomhatártól, de sokkal több, mint az IBM PC első évi forgalma (100 ezer darab). Így legfontosabb célkitűzését, hogy az IBM PC mellett egy életképes, alternatív PC-szabványt vezessen be, sikerrel teljesítette az Apple. Piaci értelemben azonban változatlanul gyengéje, hogy néhány kivételtől eltekintve, a Macintosh-ított Lisa-val sem sikerült bevennie a nagyvállalati szférát. A legtöbb vásárló a szellemi szabadfoglalkozásúak, kisvállalkozások, otthoni felhasználók és az Apple egyetemi konzorciumának résztvevői közül került ki.

Az eladások értékét tekintve az IBM a piac 35%-át kaparintotta meg 1984-ben, míg az Apple részesedés kb. 12%-ra csökkent. A kompatibilis gépekkel együtt az IBM PC konstrukció érték szerinti piaci hányada megközelítheti az 50%-ot.

A nagyvállalati szférában (ún. Fortune 1000) már a piac majd 60%-át tudhatja magáénak az IBM. A nagyszámítógépekkel (mainframe) vagy miniszámítógépekkel már rendelkező vállalatoknál – egy szelektív felmérés adatai szerint – ez az arány már valamivel kisebb, 48,7%, ha nem az értéket, hanem az alkalmazott gépek darabszámát veszik alapul. Az Apple részesedése 13,4%, a DEC cégé 5,9%, a Hewlett–Packardé 4,1% és a Tandy cégé 4,2%. Egyetlen IBM kompatibilis gyártóként a Compaq cég jutott jelentős részesedéshez a maga 4,7%-ával.

Ahol az IBM bukott

Az év nagy meglepetése volt a kisöcs PCjr bukása. Az óriási reklámkampány ellenére hatalmas eladatlan készletek halmozódtak fel az év közepére a kiskereskedelemben. A vevőknek nem kellett a nem professzionális billentyűzetével játékká degradált, a szülőtől csak robusztus méreteit öröklő új IBM-csoda.

Az IBM tulajdonképpen a saját maga által ásott verembe esett bele. PC szabványát ki akarta terjeszteni a nem vállalati alkalmazások területére is, de úgy, hogy az óhatatlanul olcsóbban piacra kerülő szabvány PC ne sérthesse a magasabb árfekvésű, igazi PC egyre növekvő eladásait.

Az Apple is égett a vágytól, hogy valahol visszaüssön az IBM-nek. Nagy tömegben piacra dobta az Apple II kompakt dobozba zárt változatát, az Apple IIc-t. A mindössze néhány kilogramm súlyú, könnyen hordozható gépet az ország kiskereskedőinek rendezett, „Apple II mindörökké" reklámozású, grandiózus party keretében vezették be a piacra.

A gép dupla nagyfelbontású színes grafikával (560 × 192 16 színben), 128 kbájt RAM-mal, 80 oszlopos kiíratási képességgel és 1 db beépített hajlékonylemezzel rendelkezett. Érthető a kereskedők lelkesedése, akik a party után 400 dollárért vihették haza az 1300 dolláros ajánlott kiskereskedelmi áron forgalomba hozott IIc első példányait. Mindjárt fel is adtak összesen kb. 50 ezer darabra szóló megrendelést. A saját ár ugyanis azt is jelezte, hogy árverseny esetén az Apple-nek még bőségesen lesznek tartalékai.

Az IBM kudarchoz az is hozzájárult, hogy az 1000 dollár körüli árfekvésű személyi számítógépek kategóriájában először kellett komolyan szembekerülnie igazi japán konkurrenciával. A többi japán gyártóval szemben meglepően mozgékony Sanyo Marubeni nagy tömegben jelent meg a piacon az automatizált gyártósorokon igen jó minőségben gyártott MBC 550-es családdal. A független termékértékelések szerint 50%-ban IBM PC kompatibilis, komplett MicroPro programcsomagokkal felszerelt gépnek kevesen tudtak ellenállni azok közül, akik olcsó professzionális gépre vágytak. A kiskereskedelem mellett igénybe vett csomagküldő (mail order) értékesítési módszer is megtette hatását.

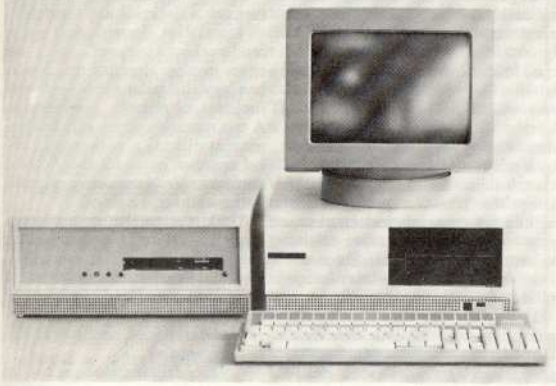
Kis gép, kis siker

Kevésbé látványos, de szintén IBM kudarc az 1984-ben piacra került „hordozható" PC vártnál kisebb keletje. A hordozhatóságot azért kellett idézőjelbe tennünk, mert a 14 kg körüli súly miatt nem is annyira hordozható, mint inkább szállítható (transportable) gépről van szó.

Az ilyen szállítható gépek piacát az Osborne nyitotta meg még évekkel ezelőtt. A cég fenomenális növekedést mutatott, 1983-ban már 160 ezer gépet adott el, és a hetedik legnagyobb személyiszámítógép-gyártó lett, mígnem az új modellek piacra hozatalánál mutatott kétbalkezes fellépés az év végére csődbe kergette. Piacát az időközben megsokasodott, új alapítású cégek minden gond nélkül átvették. A Kaypro cég már 1983-ban 113 ezer darabot forgalmazott az Osborne-énál jobb gépeiből, az IBM PC kompatibilis szállítható géppel piacra lépő Compaq cég pedig 60 ezer darabot. Ennek egy részét ráadásul Winchester-lemezes változatban.

A később megjelent „hordozható" IBM gép 1984-ben alulmaradt a Compaq-kal szemben. Egyrészt nem bizonyult olyan százszázalékosan kompatibilisnek, mint a Compaq, másrészt teljesítményével is voltak bajok. Ráadásul az IBM USA-ban működő, meghatalmazott eladóinak túlnyomó részét (1500-ból 1300-at) már 1983-ban megszerezte magának a Compaq, és különösen jó kapcsolatokat épített ki velük. Az IBM csak közvetlen értékesítési erőinek bevetésével tudta elérni, hogy megközelítse a Compaq által 1984-ben eladott 110–120 ezres darabszámot ebben a kategóriában. Az IBM saját értékesítés viszont tovább rontotta és rontja az IBM kapcsolatait a kereskedőkkel.

Az óriás cég mérsékelt sikeréhez az is hozzájárulhatott, hogy időközben a hordozható számítógépek piaca kibővült az igazi hordozható gépekkel. A folyadékkristályos vagy más lapos és könnyű megjelenítővel rendelkező, a felhasználói szoftvert ROM-ban szolgáltató és nem törlő RAM-mal (CMOS + elem) bíró gépek súlya mindössze néhány kilogramm. Így térdre is helyezhetők, minek következtében „lap size", azaz térdre helyezhető méretű gépeknek is nevezik őket. Ebben a kategóriában sikeres gyártmány a Tandy TRS–100, az Epson Geneva és a HP 110.



Tömegáru PC

Az üzleti/professzionális alkalmazások igényeinek megfelelően tervezett személyi számítógépek és a kompakt, hordozható személyi számítógépek piaca mellett egy harmadik piac is kialakult. Az ide kerülő gépeket elsősorban úgy tervezték, hogy az elmúlt évek technikai színvonala mellett minél olcsóbban, és ezzel összefüggésben minél nagyobb darabszámban lehessen őket gyártani. A gép egyes részeinek igényes kialakítása csak mint a konkurrenciával szembeni segédtényező jelentkezett, valójában minden esetben az ár bizonyult döntő tényezőnek.

Az ilyen felfogásban gyártott személyi számítógépekre ezért a tömegáru kialakítás nyomta rá a bélyegét, ami az egyes gyártóknál, illetve egyes gyártók egyes modelljeinél egyenesen az „eldobandó" (throw-away) számítógép-konstrukcióig terjedően egyszerűsítette le a személyi számítógépet.

Az 1984-ben eladott több mint 10 millió személyi számítógép túlnyomó részét az ide sorolható gépek tették ki. Piaci súlyát tekintve kétségkívül a Commodore cég a vezető gyártó. C64-es konstrukciója már majdnem professzionális alkalmazási igényeknek is megfelel, és ehhez képest az ára fantasztikusan alacsony. 1983-ban 1 millió 250 ezer darabot adtak el belőle, 1984-ben pedig az értékesítés megközelítette a kétmillió darabot.

A központi egységet állítólag 90 dolláros önköltségi áron tudja előállítani a cég. Ez és a hatalmas darabszám miatt igénybe vett tömegcikk kereskedelmi (mass merchandizing) csomagküldős és áruházi értékesítési forma lehetővé tette az eladási ár 200 dollár alá történő csökkentését. A cég eldobandó számítógép-konstrukciója, a VIC–20, jelentős szerepet játszott a nagy vetélytárs Texas kiütésében 1983-ban. Akkor több, mint egymillió darabot értékesítettek, mindössze 39 dolláros áron. Azóta ez az ár emelkedett, jól mutatva, hogy szabályos dömpingár volt.

A nagy vetélytárs Sinclair cég megközelítését a „minél nagyobb teljesítményű elektronika, minél olcsóbb köntösben" elv jellemzi. Eldobandó ZX81 konstrukciójával ez végleges formát ölt, míg a Spectrum és az 1984-ben bevezetett QL esetében mérsékeltebben jelentkezik, azaz ezeket professzionális célra is lehet használni, ha valaki megbékél a nem professzionális billentyűzettel. Eddig a Spectrum volt a cég vezető gyártmánya, állítólag több milliót gyártott belőle.

1984-ben szeretett volna áttérni a 8/32 bites QL típusra, de a bejelentés után is még javában dolgozott a konstrukción, néhány alapparamétert is meg kellett változtatnia, így az általa definiált nagy teljesítményű tömegáru PC piacot nem tudta letarolni. Ezért változatlanul kérdés, hogy az integrált alkalmazásokkal együtt árult, igen korszerű QL valóban képes lesz-e bevezetni a tömegáru megközelítést a professzionális alkalmazások területére is.

A tömegáru gyártók eddigi megközelítése egyébként is kérdésessé vált az év második felében. Megfordulni látszott a helyzet. A Commodore még azzal ütötte ki a Texas Instruments céget a nyeregből, hogy a TI gép az igényesebb kialakítás miatt drágább volt, és így nem bírta az árversenyt. Ugyanez a sors várt 1984-ben az Atarira.

Az év végére azonban úgy tűnt, hogy a vevők megelégelték ezt az árudömpinget. Igaz, hogy a fejlett tőkés országok kereseti viszonyaihoz képest elenyésző összegért, 200 dollár alatt juthattak egy alapgéphez, de bosszantotta őket, hogy egyre kevesebb perspektívát láttak a gépekben. Az eddig sikeres tömegáru PC-gyártók ugyanis minden új modellel új gépet vezettek be, a szabványokról elfeledkeztek, és úgy gondolták, hogy a régit kidobva, vegyen új gépet a vásárló. Ilyen körülmények között a játékprogramokon kívül más szoftverválaszték sem alakulhatott ki, és a gépek jelentős része egy idő után a polcra vagy a padlásra került.

NACSA SÁNDOR

Szoftver és alkalmazás

A mikroszámítógépek gyors elterjedését alapvetően meghatározta a berendezésfüggetlen és könnyen alkalmazható, sok esetben új elvű szoftvertermékek megjelenése. Mindez szoros összefüggésben áll egy új alkalmazási kultúra kialakulásával.

Szoftversiker

1983-ban a mikroszámítógépes szoftveripar forgalma először haladta meg az egymilliárd dollárt, és ebből a független szoftvergyártók 880 millió dollárral részesültek. A három első helyen ugyanakkor olyan vezető berendezésgyártók foglaltak helyet, amelyek szoftverforgalmazással is foglalkoztak:

1. IBM	110 millió $
2. Tandy (Radio Shack)	110 millió $
3. Apple	68 millió $
4. Microsoft (MS–DOS, Xenix, Multiplan)	68 millió $
5. VisiCorp (VisiCalc)	52 millió $
6. Micropro (WordStar)	50 millió $
7. Digital Research (CP/M, nyelvek)	44 millió $
8. Lotus Development (1–2–3)	38 millió $
9. Ashton–Tate (dBase II, Friday)	33 millió $
10. Peachtree (MSA) (Peachtext)	20 millió $

Ez a szoftverforgalom igen nagynak tűnik, valójában azonban a berendezések forgalmának alig 12%-át tette ki 1983-ban. Ez az alacsony részarány a magyarázat arra, hogy a berendezésgyártók eddig nem túlzottan törődtek azzal, hogy a szoftverüzletet mások bonyolítják le. Még a saját maguk által forgalmazott szoftver túlnyomó részét sem ők készítették, hanem más gyártó vagy szerző termékét adták ki. Az intenzív növekedés szakaszában sokkal nagyobb nyereségtömeget tudtak realizálni úgy, hogy csak a minél nagyobb tömegű berendezésgyártásra koncentráltak. Mindez egyértelmű magyarázatot ad az IBM eddigi stratégiájára, illetve arra is, hogy miért volt az Apple Lisa-kísérlete szükségszerűen bukásra ítélve.

Alkalmazás-orientáltság

A mikrogépes szoftvertermékek piaca három irányban fejlődött, és a három irány időbeli piaci felfutása is megfelel nagyjából a felsorolás sorrendjének:

1. Szoftverfejlesztést és programozást támogató termékek, mint operációs rendszerek, programozási nyelvek és segédprogramok

Amíg az Apple eléggé házon belül tartotta ezt a piacot, addig a többi berendezés esetében a gyártótól független operációs és nyelvi rendszerek lettek a meghatározók. Az IBM MS–DOS melletti döntése a CP/M-et és más rendszereket egyre inkább háttérbe szorította. A Digital Research is az IBM PC–DOS-szal való kompatibilitásra törekszik CP/M rendszerének továbbfejlesztésében. Programozási nyelvek tekintetében jó minőségű és teljes választék alakult ki.

A Digital Research igen jó eredményeket ért el a különböző hardver operációs rendszer architektúrákon egységesen alkalmazható nyelvi rendszerek tekintetében. A jövőt a többfelhasználós nyílt rendszerszoftver-szabványok (Unix, Xenix, illetve DR-Net, MS – Net) elfogadása vagy a berendezésgyártók által diktált szabványok fogják meghatározni. Folyamatban van a grafikai interfészek szabványosítása is.

2. Közvetlen alkalmazási termékek

Ezeknek a termékeknek két csoportja van. Az elsőbe tartoznak az általános üzletviteli gyakorlat egy-egy célfunkcióját, illetve azok valamilyen együttesét támogató programcsomagok. Ilyen célfunkciók: a könyvtartozások és követelések vezetése, a megrendelésfelvétel és számlázás, a bérelszámolás, a főkönyv vezetése stb.

A második csoportba azok a közvetlenül alkalmazható programok tartoznak, amelyek egy-egy speciális alkalmazási terület (iparág, szakma) igényeit elégítik ki. Jellegzetes területek: biztosítási üzletkötés, ingatlanközvetítés, szállodaipar (beleértve az utazási irodákat), a közjegyzői munka, az újságárusítás, az épületek műszaki felmérése stb.

3. Alkalmazásfejlesztést támogató termékek

Ezek a termékek egy-egy általános alkalmazási funkciót vagy azok valamilyen együttesét támogatják, meglehetősen általános módon. Ide tartoznak az adat- és információfeldolgozás adatbázis-kezelő funkciói; a számítógéppel támogatott döntéshozatal feladatainak igények szerinti meghatározását biztosító elektronikus feladatlap (electronic spreadsheet) és üzleti grafika (business graphics) funkciók; a számítógépes dokumentumfeldolgozás feladatait a konkrét igények szerint lehetővé tevő szövegfeldolgozó (word processor) funkciók; a különböző számítógépes adatrendszerek, tipikusan mikro- és nagyszámítógép közötti általános adatkommunikáció funkciói; az egyszerűbb gép–gép közötti kapcsolat képernyő-kommunikációs funkciói stb.

Minél több általános alkalmazási funkciót minél általánosabban támogat az ilyen termék, annál szélesebb körben használható az alkalmazások *nem* gépi utasítások hosszú sorozatának leírásával történő fejlesztésére. Ezek a termékek tehát a hagyományos programozási nyelvekben való programfejlesztést váltják fel. A fejlődési tendencia az egyes alkalmazási funkciók minél általánosabb szolgáltatása, a felhasználói interfész hatékonyabbá tétele, a programozás nélküli alkalmazásfejlesztési rendszer nyílttá és bővíthetővé tétele, valamint egyre több általános alkalmazási funkció hatékony integrálása.

Egy 1984-ben készült felmérés szerint a nagyvállalati szférában működő mikroszámítógépek 80%-án használtak rendszeresen elektronikus feladatlapot, 66%-án szófeldolgozót, 43%-án adatbázis-kezelőt, 53%-án aszinkron terminálkommunikációt, 20%-án mikro-nagyszámítógép adatkommunikációt és 8%-án üzleti grafikát. Már 1983-ban csak azok a független szoftvergyártók lehettek az első tíz között (10 és 65 millió dollár közötti éves forgalom), amelyek *nem* a közvetlen alkalmazási szoftverekben látták legígéretesebbnek az alkalmazási szoftver piacot.

Alkalmazásgenerátorok

A piacon alkalmazásgenerátorként hirdetett termékek technológiája a mikroszámítógépektől függetlenül, a nagyszámítógépes technika fejlődésének eredményeképpen alakult ki.

Az alkalmazásgenerátoroknak három technikai generációja van. Az ún. második generációs alkalmazásgenerátorok jellegzetessége, hogy általánosított fájlkezelő rendszerekre épülnek rá, ezért a lehető legtöbb korlátozással bírnak és a lehető legkevésbé általános a használhatóságuk. A harmadik generciós generátorok valamilyen igazi adatbázis-kezelő rendszerrel szorosan egybeintegrált rendszerként jelentkeznek. Végül a negyedik generációs alkalmazásgenerátorok egy általános adatmodell szerinti általánosítással kialakított adatbázis-kezelő rendszer elválaszthatalan részei, és bátran lehet őket negyedik generációs programozási nyelveknek nevezni, mivel igen magas, az alkalmazási feladat specifikációs szintjén teszik lehetővé a fejlesztést, a korábbi nyelvek ún. eljárás-orientáltságával ellentétben.

Az alkalmazásgenerátorokban közös felismerés, hogy az adat- és információfeldolgozásban a következő általános alkalmazási funkciók vannak: adatbázis-definíció; adatbázis-lekérdezés (query); adatbevitel (data entry); adatkarbantartás (update); tabló (jelentés) előállítás (report generation).

Az alkalmazásgenerátorok ezeknek a funkcióknak megfelelően kialakított, összehangolt nyelvi rendszerként jelennek meg. Az egyes nyelvi alrendszerek általában egy, de néha több összevont funkciót együttesen támogatnak. A mikroszámítógépes alkalmazásgenerátoroknál a parancsnyelvekkel szemben egyre inkább előtérbe kerülnek az interaktív, vizuális kommunikációs nyelvek.

Tipikus személyi számítógépes termék a dBase II, ennek javított változata a dBase III, valamint a Personal Pearl. Ezek csak egyszerű alkalmazásgenerálási képességgel rendelkeznek. A személyi számítógépes adatbázisok azonban általában egyszerűek.

A többfelhasználós mini- és nagyszámítógépes technikából „mikrósított" alkalmazásgenerátorok jó példája az Oracle cég hasonló nevű terméke és az Information Builders cég Focus nevezetű alkalmazásgenerátora. Ezek a negyedik generációs nyelvek egyfelhasználós, személyi számítógépes változatban is piacra kerültek a legutóbbi időben. Így igen nagy méretű bonyolult adat- és információfeldolgozási feladatokat is meg lehet ma már oldani személyi számítógéppel anélkül, hogy hosszadalmas BASIC vagy COBOL kódolást kellene folytatni.

Mivel ezekkel az alkalmazásgenerátorokkal még a nagy és bonyolult feladatok fejlesztése is ti-

zedannyi idő alatt elvégezhető, valamint nem utolsósorban nem kell programozókat és rendszerszervezőket bevonni a fejlesztésbe, a mikroszámítógépek alkalmazásának egészen új irányát nyitják meg ezek a rendszerek. Egy tipikus Oracle alkalmazás például egy építkezés helyszíni munkálatainak irányítása IBM PC-n megvalósított Oracle alkalmazással, míg a teljes irányítási feladat egy központi VAX szuperminin működő Oracle rendszeren fut.

Generikus alkalmazások

A piacon generikus alkalmazásként emlegetett termékek a személyi számítógépes technika szülöttei. Igen jelentős szerepük van a személyi számítógépek elterjedésében, mivel az adott általános alkalmazási funkció, illetve funkciócsoport által meghatározott alkalmazások egész fajját támogatják. Tulajdonképpen innen ered a termékcsoport közös megnevezése (genus latinul fajt jelent).

Egy tipikus generikus alkalmazás a szófeldolgozó (word processor). Nagyon sokféle termék sorolható ebbe a termékkategóriába. A közös bennük az, hogy az ún. dokumentumfeldolgozással kapcsolatos funkcióhalmaz egy részhalmazát támogatják az egyes termékek.

Dokumentumon (document) szöveges és illusztrációs részek szerkesztett együttesét kell értenünk. A szöveges rész bevitele során a személyi számítógépnek a szavak megfelelő elhelyezésével kell foglalkoznia (például a szavak átvitele a következő sorba, szavak helyes elválasztása automatikusan stb.). A dokumentum vagy szöveges részeinek szerkesztése során szintén a szavak megfelelő átrendezése a személyi számítógép feladata. Innen ered a termékek megnevezése. Tipikus szófeldolgozó csomag a WordStar, a Microsoft Word, a Multimate stb.

A dokumentumfeldolgozás persze igen összetett funkcióhalmaz. Ide tartozik a szövegfeldolgozás összetett funkciója, beleértve a szöveg bevitelét adott lapformátum szerint, a szöveg átdolgozás közbeni, illetve a nyomtatást megelőző szerkesztését, a helyesírás ellenőrzését (spelling check) mint a legfőbb részfunkciókat.

A táblázatok, grafikus és más illusztrációk előállítása, szerkesztése és beillesztése a szöveg megfelelő helyeire, szintén dokumentumfeldolgozási funkció. Ezt tipikusan nem a szófeldolgozó program végzi, hanem az elektronikus feladatlap, az üzleti grafika vagy az általános illusztrációs grafika (amilyen például a MacPaint), tehát egy másik generikus alkalmazás. A szófeldolgozónak csak az adatkommunikációra kell felkészítve lennie ezzel a másik generikus alkalmazással.

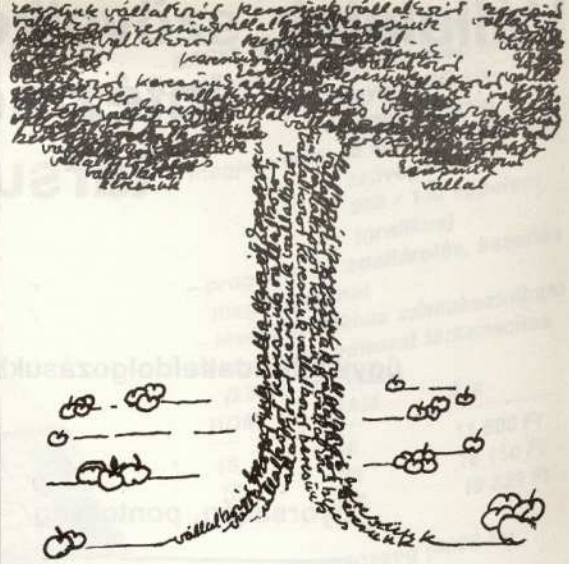
Végül vannak a dokumentum, mint komplett mű kezelésével kapcsolatos speciális funkciók, például a fej- és/vagy lábléc készítése minden lapon, a tartalomjegyzék, a név- és tárgymutató készítése, vázlat (outline) készítés a dokumentum írása előtt stb. Ezeket a funkciókat már kezelhetik a szófeldolgozók, de tipikusan csak azokban a szófeldolgozókban találjuk meg, amelyek az ún. hosszú dokumentumok előállítására vannak szakosítva. Feljegyzések készítésénél nyilván nincs szükség ezekre a lehetőségekre, viszont egy író feltétlenül igényt tart rájuk. Szakképzett leírók és titkárnők esetében is más kialakításra van szükség, mint alkalmi gépelők esetében.

Az elektronikus feladatlap és üzleti grafika termékcsoportok amellett, hogy megfelelő táblázatos illusztrációt szállítanak a dokumentumfeldolgozáshoz, alapvetően a döntéshozatal támogatásával (decision support) kapcsolatos funkciócsoporthoz tartozó általános alkalmazási funkciók. Az elektronikus feladatlap a táblázatba foglalt adatok és a közöttük meghatározott függvényszerű összefüggések modellezési nyelve, az üzleti grafika pedig a táblázatos adatok grafikus ábrázolásának nyelve. A két terméket egyre inkább integrálják egy termékbe, és akkor a döntéshozatal támogatási (modellezési) alkalmazások generátorát kapjuk. Tipikus termékek: a VisiCalc, a Multiplan, a Supercalc, a VisiTrend, a VisiPlot stb. Tipikus felhasználók: a vezetők és a gazdasági szakemberek.

Csak a legfontosabb generikus alkalmazásokat említhettük e rövid, felsorolásszerű ismertetés keretében. Még azt kell felvillantanunk, hogy hogyan is jutunk el a generikus alkalmazás használata során a konkrét, egyedi alkalmazásig. Általános megfogalmazásban: az alkalmazási feladat konkrét specifikálásával a generikus alkalmazás által nyújtott általánosított fogalmi rendszerben. Konkrétabban: a dokumentum konkrét jellemzőinek meghatározásával, az elektronikus feladatlap „kitöltésével", a konkrét illusztráció „megrajzolásával" (MacPaint) stb.

A lényeg minden esetben ahhoz hasonló, amit az alkalmazásgenerátoroknál láttunk: a számítógép konkrét igények szerinti alkalmazhatóságához nem kell hagyományos programozást folytatnunk. A generikus alkalmazások felhasználási területén a programozási módszerrel maga a konkrét alkalmazhatóság vált volna kérdésessé. A generikus alkalmazások szerepe tehát még fokozódni fog a jövőben, és ezek fogják az alkalmazási kultúrát is meghatározni.

NACSA SÁNDOR



Vállalkozót keresünk

SZÁMÍTÁSTECHNIKAI SZOLGÁLTATÓK KATALÓGUSÁ-nak

egyszeri összeállítására és folyamatos karbantartására

A katalógus nyújtson egyszerű tájékozódási lehetőséget a számítástechnikai tevékenységi körökhöz rendelten

- **a szolgáltatók által eddig elvégzett tevékenységről,**
- **a referenciahelyekről,**
- **az értékesíthető, ill. az adaptálható munkákról,**
- **a kapacitásadatokról, ill. a vállalkozási készségről.**

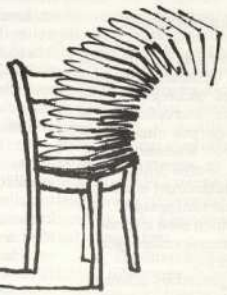
A katalógus összeállítására vonatkozó ajánlatokat, a tartalmi, formai kivitelre és a vállalkozás körülményeire vonatkozó elképzeléseket kérjük a 1502 Budapest, Pf. 129 címre beküldeni.

ÉLELMISZERIPARI ÜGYVITELSZERVEZÉSI ÉS GÉPI ADATFELDOLGOZÓ VÁLLALAT

Telex: 22-6035

Vállalatok, szövetkezetek, intézmények, társulások, figyelem!

Hívjanak,
mi megoldjuk
ügyviteli adatfeldolgozásukban jelentkező
gondjaikat



Ne feledjék:
gyorsaság, pontosság = hatékonyság

A PRODUKTORG Szervezési Vállalat Számítástechnikai Fejlesztési és Kisszámítógép-Alkalmazási Irodája MO8X, Proper–8, TAP 34 és egyéb mikrogéprendszerekre adaptálható programcsomagokkal és programokkal áll rendelkezésükre. A programok általános jellegűek, így azok az Önök kívánságaira, szükségletük szerint alakíthatók.

Bevezetésre ajánljuk a következő programcsomagokat és programokat:

- Főkönyvi könyvelést támogató programcsomag
- Értékesítési programcsomag
- Személyzeti nyilvántartás programcsomag
- Állóeszközök analitikus nyilvántartása programrendszer
- Rendelésnyilvántartás (szállítók) programrendszer
- Raktárgazdálkodási programrendszer
- Munkaügy és bérszámfejtés programrendszer
- Elő- és utókalkulációs programrendszer
- MICROOPT lineáris programozási programcsomag, matematikai modellen alapuló optimumszámítások (pl. gazdaságos termékösszetétel)
- Termelékenység- és hatékonyságvizsgálat

Várjuk szíves érdeklődésüket, készséggel állunk rendelkezésükre a 350-169, 154-090/144, 178, 323, 408, 661 telefonszámokon, ahol munkatársaink részletes felvilágosítással szolgálnak.

PRODUKTORG
SZERVEZÉSI VÁLLALAT

PRODUKTORG Szervezési Vállalat
Számítástechnikai Fejlesztési és
Kisszámítógép-Alkalmazási Iroda
1251 Budapest II., Fő u. 68.

Tanuláshoz,
munkához, játékhoz pro primo:

MICROKEY

PRIMO

személyi számítógép

– BASIC programozási nyelv
– billentyűzet: kapacitív elven működő magyar ABC kis- és nagybetűkkel
– megjelenítés: 16 sor × 42 betűhely szöveg
256 × 192 képelem (grafikus)
– program- és adattárolás, kazettás magnetofonnal
– tévékészülékhez csatlakoztatható
– háromféle változat tárkapacitás (kbájt) szerint

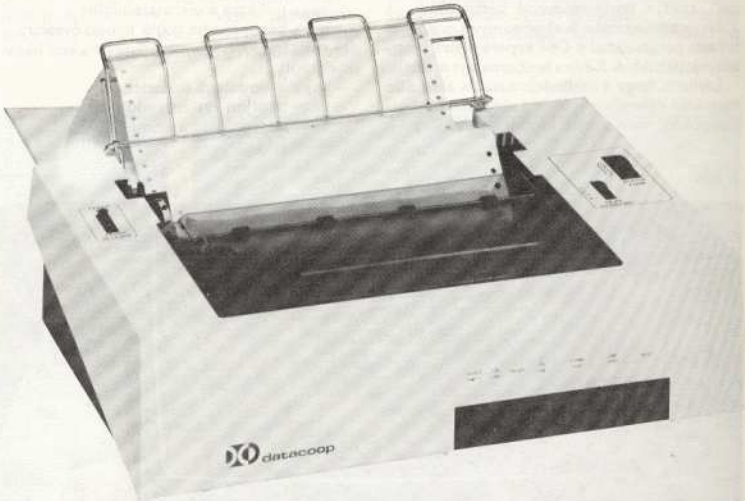
ROM	RAM	Ára
16	16	11 500 Ft
16	32	16 150 Ft
16	48	19 339 Ft

– saját tápegység (4600 Ft)

DCD–PRT–80

datacoop

grafikus mozaiknyomtató



– 80 karakter/s nyomtatási sebesség
– 80 karakter/sor
– 9 × 7 pontos mozaikkarakter
– kétirányú nyomtatás logikai kereséssel
– mikroprocesszoros felépítés
– traktoros vagy gumigörgös papírtovábbítás
– változtatható papírszélesség (115–224 mm)
– Centronics vagy DZM interfész

EMO ELEKTROMODUL BUDAPEST

Forgalmazza:

MAGYAR ELEKTROTECHNIKAI ALKATRÉSZKERESKEDELMI VÁLLALAT

2. sz. bolt
Budapest XIII.,
Jászai Mari tér 5.
T.: 530-800

Klubújdonságok

Ebben a rovatban a HCC és esetleg más klubok által kidolgozott, szélesebb kör számára is érdekesnek tűnő hardver- és szoftverújdonságokat ismertetek. Várom tehát a klubok ilyen eredményeit ismertető cikkeket.

Lapunk előző számában már beszámoltam arról, hogy a HCC Commodore Szekciói (64 és VIC–20) ún. klublemezeket hoznak forgalomba, és ismertettem az eddig meglevő programokat. Megemlítettem azt is, hogy a lemezek használatához szükséges hardvereszközöket a lemezekhez tartozékként adjuk. Néhány tartozékot (botkormányt, csatlakozót) fel is soroltam.

Mivel sok Commodore-használó nem rendelkezik lemezegységgel, a programok közül azokat, amelyeknél ez egyáltalán lehetséges, kazettán is forgalmazni fogjuk. Amennyiben valamelyik vidéki szervezet társulni akar ehhez az akcióhoz, szívesen látjuk, és várjuk programjait.

A lemezekhez járó botkormány saját konstrukciójú, *mindenféle számítógéphez használható, a szokásosnál sokkal nagyobb élettartamú, olcsó eszköz*. Eddig a Commodore gépeken kívül eredményesen kapcsoltuk össze Primóval és Dragonnal is. Az *1. képen* a Primo képernyője látható a rajzolóprogram egy részével és „rajzzal", amit a botkormánnyal készítettünk. A *2. képen* ugyanazzal a botkormánnyal és a KoalaPaint programmal a C64 képernyőjére készített rajz látható. A *3. kép* a botkormányt mutatja.

Látható, hogy a szabadalmaztatás alatt álló megoldás külsejében is eltér a szokásos botkormányoktól, mert sokkal kisebb átmérőjű, amit a kis helyigényű megoldás tett lehetővé. (A szerző egyébként 50 forint jelképes összegért bármilyen gyári botkormányt átalakít Primóhoz használható botkormánnyá.)

Mind VIC–20, mind C64 típusú gépekhez, kellő számú igény beérkezte esetén, készítünk tárbővítőket 256 k címterületig, párhuzamos porton keresztül dolgozó, tehát a szokásosnál sokkal gyorsabb lemezillesztőt.

Az új programok közül kiemelkedő jelentőségű és értékű egy programvizsgáló professzionális célú fejlesztőprogram. A szokásos assembler-disassembler, tárvizsgáló-módosító-kiíratő, regisztermódosító-kiíratő, programfuttató funkciókon kívül számos jól használható paranccsal rendelkezik:

- beállítható szubrutinszint mélységű nyomkövető, amely minden lépés után kiírja a regiszterek tartalmát és az utasítást assembler nyelven,
- n-utasítás szimulálása, regiszterkijelzéssel vagy anélkül,
- az utolsónak végrehajtott n-utasítás kiíratása assembler nyelven,
- töréspont-beiktatás a szimulációba úgy, hogy a töréspontnál
  - írja ki a regiszterek tartalmát,
  - vagy nullázza a ciklusszámlálót,
  - vagy kapcsolja be (ki) a nyomkövetést,
  - vagy írja ki, hogy megadott címeken hányszor futott át,
  - vagy írjon ki egy üzenetet,
  - vagy ugorjon egy megadott címre,
  - vagy kapcsolja ki a szimulálást,
  - vagy vizsgálja meg egy adott feltétel teljesülését,
  - vagy írja ki a töréspont eléréséig szükséges ciklusszámot,
- töréspont-kiíratás, úgy, hogy kiírható
  - az összes,
  - vagy az adott címre ugrások címei,
  - vagy egy adott címtartományra ugrások címei,
- a verem n-legfelső szintjének kiíratása,
- bájt, ill. karaktersorozat-kikereső (legfeljebb 80 bájt),
- ciklusszámlálás legfeljebb 100 milliárd ciklusig.

A szimuláció nem módosítja a valóságos tárterületet. Ez a program más 6502 típusú mikroprocesszoros gép esetén is, például Apple II-nél is használható.

A C64 CP/M rendszere egyetlen lemezegységet kezel. A Klub vállalja CP/M lemezeknek két lemezegységet kezelővé alakítását.

DR. SIMONYI ENDRE

1. kép

2. kép

3. kép

Építsünk számítógépet! VII.

Az előző részben ismertettem a videokártya videofunkcióit, a CPU-kártyával történő kapcsolattartást, a video–IC címzését, a képernyőtár címzését, a tárak kapcsolási rajzát. Most nézzük a kártya további részeit.

Az előző számunkban közölt VI. rész *4. ábráján* látható és a címzés szempontjából fontos VA1Ø és VA11 jeleket az *1. ábrán* látható módon állítják elő az IC 21, 22, 23 (74157) multiplexerek. Ezek vagy a video–IC címeit kapcsolják a memóriák címvonalaira, vagy a busz címeit, amikor a videomemóriák író/olvasó kapcsolatot létesítenek.

A képernyővel kapcsolatos műveletek alatt a video–IC folyamatosan változtatja a videomemória-címeket, ha szinkron jeleket küld, ha kioltó jeleket küld, ha a karaktereken belüli címeket állítja be stb. Az adatvonalakra a grafika vagy az EPROM-ban tárolt karaktergenerátor ad jeleket.

Az EPROM (IC 41) kiválasztása, illetve kiiktatása a PIA (IC 33) PBØ adatvonalával történhet. Ha az IC 23A 7. lábán „1" jel van, akkor az EPROM van kiválasztva. Az EPROM 20. lábára magas értéket adva, a kimenetein nem lesz jel. Az IC 29, 3Ø (74367) képes a grafikus jelek továbbengedésének megakadályozására. A képpontforrás mindkét esetben közös, és az IC 39, 4Ø (7486) bemenő jeleit szolgáltatja. Az ismertetett körök a *2. ábrán* láthatók.

A videoáramkörök további részeit a *3. ábra* mutatja. A képpont-órajelet egy párhuzamosan kapcsolt kristályból, az IC 37 (74Ø4) három kapujából és passzív elemekből álló kör állítja elő. Az IC 36 (74163) az órajelet nyolcad, illetve tizenhatodrész frekvenciájú jelre osztja. A kétféle leosztás teszi lehetővé az alfanumerikus és a grafikus üzemmód létezését. A 256 × 192 képpontos grafika soronként 256, a 64 × 16 karakteres alfanumerikus üzemmód soronként 512 képpontot használ, ezért ez utóbbi kétszeres frekvenciával dolgozik.

A vezérlést az IC 23A (74157) végzi PBØ jel segítségével. A jel az IC 35 (74ØØ), IC 34 (74LS74) segítségével kerül a léptetőregiszterbe. A vibrálásmentes kép előállítására szolgál az IC 27 (74Ø4) által invertált M-jellel vezérelt IC 47 (74LS74), amivel az IC 34 (74LS74) működését befolyásoljuk. A karaktersebesség órajelét visszavezetjük a video–IC-be. Ez lépteti a következő karakterre a kiíratást, amikor már a karakter minden sora a képernyőre került.

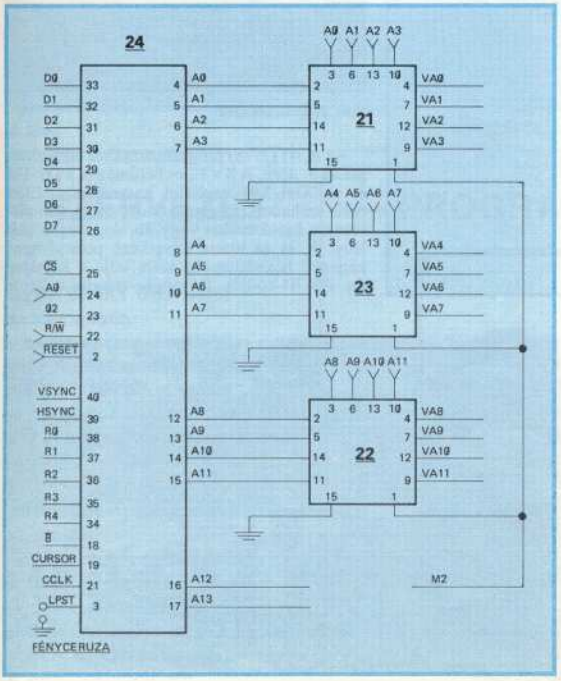
A léptetőregiszter, az IC 28 (74166) gondoskodik arról, hogy megfelelő számú képpontpozíció íródjon ki karaktersoronként. Ezek a pontok KIZÁRÓ–VAGY kapcsolatba kerülnek a fénypontal. Ezt az IC 32 (7486) valósítja meg. Így a fénypont inverz-kiválasztó hatást gyakorol a karaktercellára. Az impulzusok az IC 35 (74ØØ) segítségével összekapuzódnak a kioltó jellel. Egy átkötéssel elérhető, hogy a vízszintes vonalak pontokból és ne vonalakból álljanak. Ez a fényerősség és kontraszt újraszabályozását igényli a kijelzőn. Jó minőségű monitorokon ez a mód sokkal jobb képminőséget eredményez.

Lehetőség van arra is, hogy külső kioltás segítségével elkerüljük a képernyő villogását, addig, amíg adatátvitel van a képernyő tárában. Erre szolgál az Ā-jel. Az IC 35 jelét invertáljuk az IC 37 (74Ø4) segítségével és keverjük az IC 29 (74367) és IC 16 (7432) által a komponensekből előállított szinkronjellel, melyet még az IC 15-tel (7432) invertálunk. Ha az IC 15 bemenetét nem az IC 16 kimenetére, hanem földre kötjük, akkor TTL-szintű vízszintes és függőleges szinkronjeleket kapunk. Az IC 32 6. lábán TTL-szintű összetett videojelet kapunk.

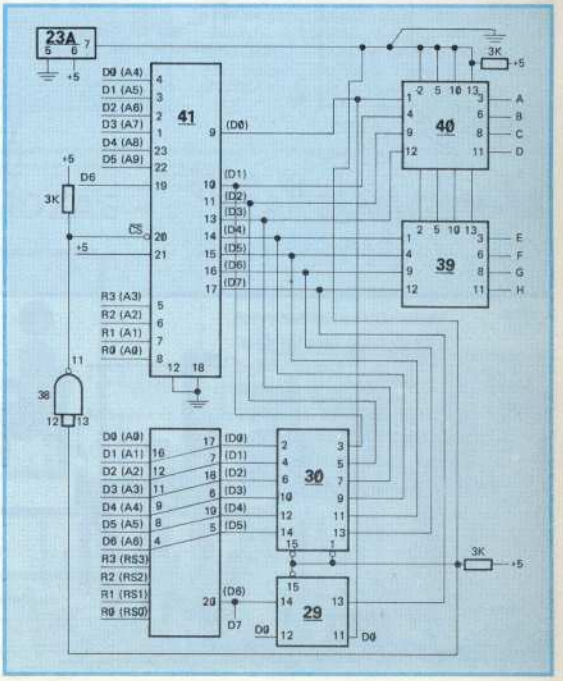
Az 1. ábrán láthatjuk, hogy fényceruza használatára is lehetőségünk van.

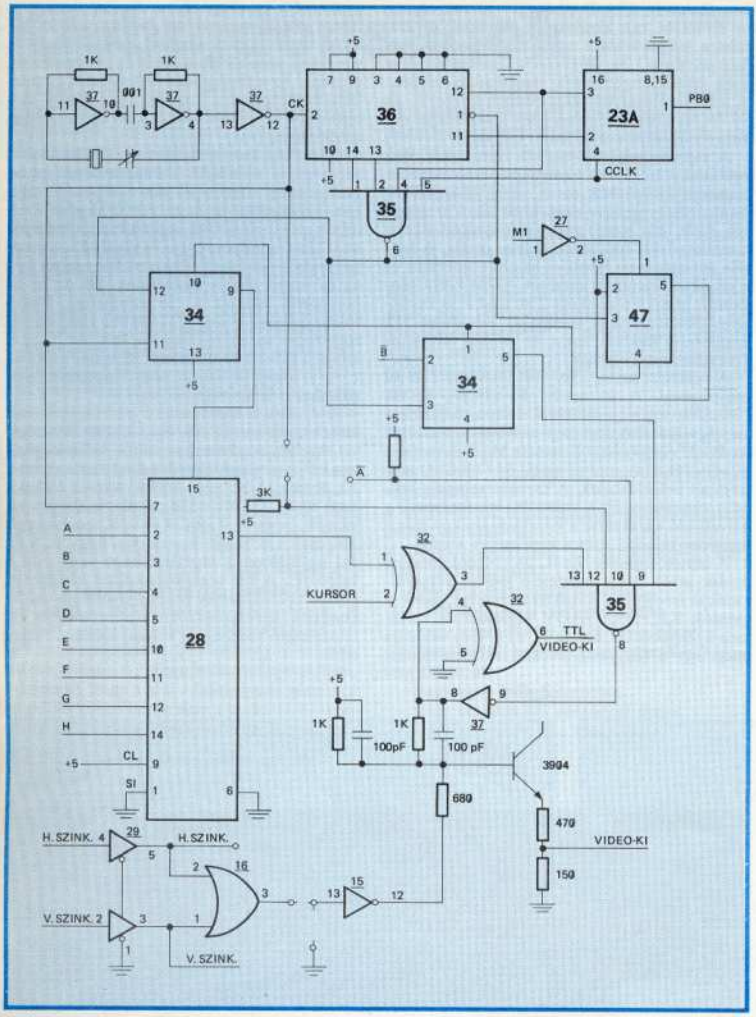
A *4. ábra* a kártyán levő párhuzamos port kialakítását szemlélteti. Egy átkötés segítségével, ha bemenetként használjuk, lehetőségünk van külső megszakításnak a rendszerbe vitelére. Az ábrán 11 be nem kötött vezeték látható. Ezek az IC 33 (MC 68Ø) adatlapja alapján azt jelentik, hogy az IC ún. A oldalát nem használjuk, csak a B oldalt. Ennek oka, hogy az A oldal ugyanazon a memóriacímen van, mint a video–IC. A két vezérlővezetékkel kiegészített portot használhatjuk például nyomtató csatlakozására, lemezillesztő készítésére, további kazettaillesztő készítésére, billentyűzetillesztőként stb.

1. ábra



2. ábra





3. ábra

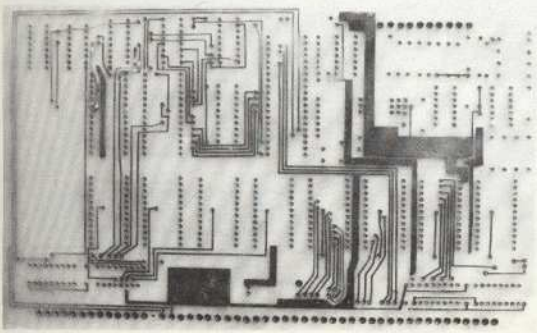
KÓD	KARAKTER	KÓD	KARAKTER	KÓD	KARAKTER
ØØ	NUL	2B	+	56	V
Ø1	SOH	2C	,	57	W
Ø2	STX	2D	-	58	X
Ø3	ETX	2E	.	59	Y
Ø4	EOT	2F	/	5A	Z
Ø5	END	3Ø	Ø	5B	(
Ø6	ACK	31	1	5C	\
Ø7	BEL	32	2	5D	)
Ø8	BS	33	3	5E	^
Ø9	HT	34	4	5F	_
ØA	LF	35	5	6Ø	`
ØB	VT	36	6	61	a
ØC	FF	37	7	62	b
ØD	CR	38	8	63	c
ØE	SO	39	9	64	d
ØF	SI	3A	:	65	e
1Ø	DLE	3B	;	66	f
11	DC1	3C	<	67	g
12	DC2	3D	=	68	h
13	DC3	3E	>	69	i
14	DC4	3F	?	6A	j
15	NAK	4Ø	@	6B	k
16	SYN	41	A	6C	l
17	ETB	42	B	6D	m
18	CAN	43	C	6E	n
19	EM	44	D	6F	o
1A	SUB	45	E	7Ø	p
1B	ESC	46	F	71	q
1C	FS	47	G	72	r
1D	GS	48	H	73	s
1E	RS	49	I	74	t
1F	US	4A	J	75	u
2Ø	SP	4B	K	76	v
21	!	4C	L	77	w
22	"	4D	M	78	x
23	#	4E	N	79	y
24	$	4F	O	7A	z
25	%	5Ø	P	7B	{
26	&	51	Q	7C	/
27	'	52	R	7D	}
28	(	53	S	7E	~
29	)	54	T	7F	DEL
2A	*	55	U		

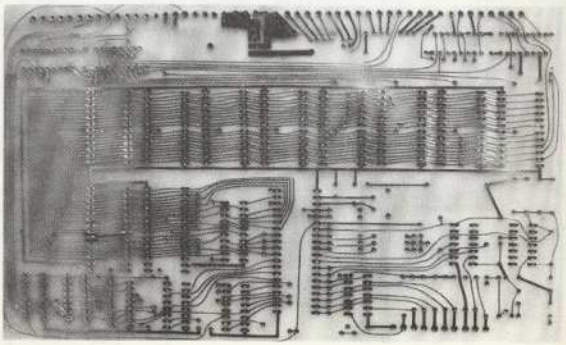
A fontosabb kontrolkarakterek jelentése:

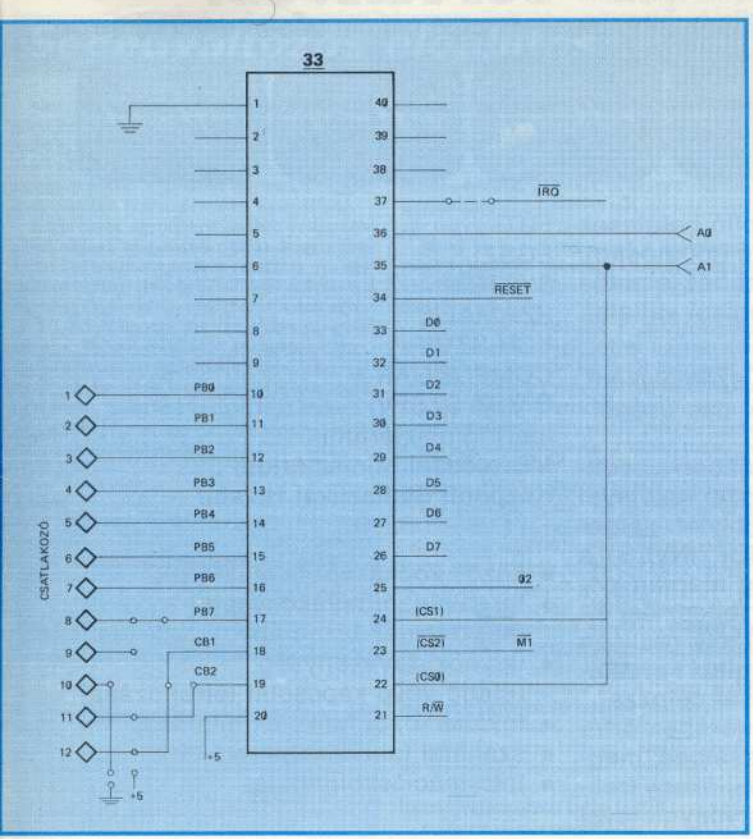
BEL	– hangadás
BS	– visszaléptetés
ØA	– soremelés
ØD	– kocsi vissza
1B	– törlés

ASCII-kódtáblázat

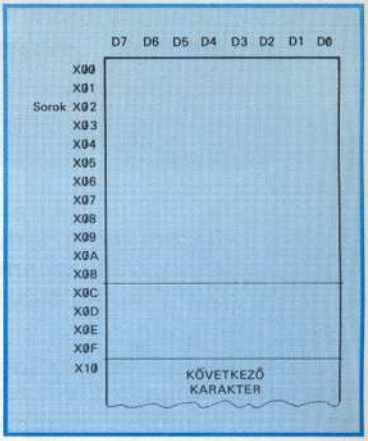
Az IC 41 (2716) programozásához ad segítséget az *5. ábra*. A 8 × 12-es betűmátrix a D7–DØ és az XØØ–XØF területet használhatja. Ezen belül szabadon programozható, és így igen változatos karaktereket vagy ábrákat hozhatunk létre. X és az utána következő pozíció mint kétjegyű hexadecimális szám adja a karakter ún. ASCII-kódját. DØ értéke mindig Ø, így a







4. ábra



5. ábra

karakterek között egy kis közt állítunk elő. Az utolsó négy sor a sorköz beállítása céljából szintén Ø.

Mivel 16 bájtot használunk fel egy karakter előállítására, összesen 128 féle karaktert tudunk generáltatni. Az *ASCII-táblázatból* láthatjuk, hogy az első 32 ebből ún. kontrollkarakter, aminek nem felel meg látható karakter a képernyőn. Amennyiben tudni akarjuk, hogy milyen kontrollkarakterrel dolgozik a gép, ezek helyére például rövidítéseket írhatunk, lekicsinyített formában, ami által ezek a jelek könnyen megkülönböztethetők lesznek a többi karaktertől. A kártyán – a kapcsolás csekély változtatásával – az IC 41 lehet 2732 vagy TMS 2532 típusú EPROM is. Ebben az esetben már 256 féle karaktert tudunk megvalósítani.

DR. SIMONYI ENDRE

SZERVEZÉS

Az INTERAKT GMK saját számítógépén – bérmunkában – vállalja a rendszeres adatfeldolgozást adatelőkészítéssel együtt elsősorban az alábbi területeken;

- **Anyagkönyvelés, készletgazdálkodás,**
- **Szerződés és rendelés-nyilvántartás,**
- **Álló- és fogyóeszköz-nyilvántartás,**
- **Szállító- és vevő folyószámla-könyvelés,**
- **Számlaellenőrzés,**
- **Bér- és munkaügyi nyilvántartás,**
- **Bérelszámolás stb.**

– PROGRAMOZÁS –

INTERAKT
Szervezési és Számítástechnikai
Gazdasági Munkaközösség
Bp. II., Fazekas u. 10–14. V/46.
389–897
Közös képviselő: Pék Ágnes

SZAKTANÁCSADÁS

A nyilvántartások folyamatos vezetésével lekérdezési lehetőséget biztosítunk telefonon is. Vállaljuk továbbá működő rendszerek korszerűsítését, különböző mintarendszerek adaptálását. Érdeklődni lehet a 389-897 sz. telefonon.

Kedves Érdeklődők! Tájékoztatjuk, hogy 1985. január 1-ével

megalakult a Szoftverkereskedelmi és Fejlesztési Betéti Társulás.

SOFTinvest

1525 Budapest Pf. 51. Tel.: 358-530/866

Az alapító betétesek három minisztérium (főhatóság), valamint hét jelentős számítástechnikai vállalat.

FÜTI
ÉGSZI
SYSTEM
SZÁMALK
KSH SZÜV
COMPORGAN
OKISZ SZSZV
Ipari Minisztérium
Művelődési Minisztérium
Központi Statisztikai Hivatal

A SOFTINVEST önálló jogi személyként, vállalati gazdálkodási rendben, nyereségérdekelt szervezetként működik.

Tevékenységének középpontjában a szoftver, mint termék, áru áll. Saját szervezetével fejlesztői tevékenységet *nem* folytat. Jelentősnek mondható tőkéjével a SOFTINVEST a közös kockázat és érdekeltség alapján potenciális partner minden olyan fejlesztési elképzelés megvalósításában, amely reálisan becsült – hazai vagy külföldi – piaci keresletet kielégítő szoftver létrehozását célozza.

Kereskedelmi tevékenységi körében értékesítésre átvesz – speciális esetben megvesz – szoftvereket, azokat hajlékony reklám-propaganda eszközök igénybevételével forgalmazza.

Folyamatos piackutatást végez a szoftverkereslet megismerése érdekében, különösen azokon a területeteken, ahol a keresletet egy új géptípus, vagy egy új, korábban számítástechnikával nem kielégíthető szervezési igény megjelenése keltette. A SOFTINVEST rugalmasan kíván reagálni, gyors külső kapacitások felhasználásával.

A SOFTINVEST a számítástechnikai eszköztár minden eleméhez kapcsolódó szoftver létrehozásában, forgalmazásában érdekelt, kiemelt szerepet biztosítva a mikroszámítógépeknek.

A mikrogépes – különösen a nagyobb számban felhasználásra kerülő – szoftver értékesítése jelentős előszervezési igényt támaszt. Ennek kielégítésére – tartós kapcsolattal – keresünk szervezői kapacitásokat a kisvállalkozók köréből.

- közös kockázatú fejlesztések
- forgalmazói tevékenység (késztermék bizományosi értékesítések)
- ügynöki tevékenység (eladó-vevő kapcsolat létrehozása)
- fővállalkozói funkciók ellátása,
- szakmai tanácsadás
- információszolgáltatás

Minden piacképes SZOFTVER termék forgalmazásában, vagy ötlet megvalósításában partnerek vagyunk

A potenciális számítástechnikai alkalmazók információigényének kielégítésére tájékoztató, tanácsadó tevékenység beindítását tervezzük, amelynek ellátására külső kapacitást kívánunk igénybe venni.

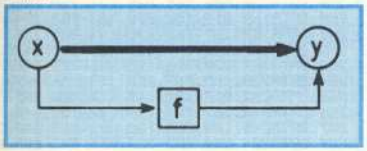
Reméljük, hogy sikerült bemutatni a SOFTINVEST tevékenységét, felkeltenünk érdeklődését oly módon, hogy valamely témában már a közeljövőben partneri kapcsolatra léphetünk.

Segédváltozós előállítás

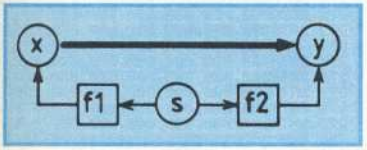
Nem egyértékű függvényeknek, illetve ezek görbéjének előállítását segédváltozó alkalmazásával gyakran könnyűvé tehetjük. A segédváltozós előállíthatóság azonban nincs a többértékűséghez kötve; a módszer egyértékű függvények esetében is eredményesen használható.

Függvények egymáshoz tartozó értékpárjai előállításának megszokott útja az, hogy x értékéhez kiszámítjuk y értékét *(1. ábra)*. Az egymáshoz tartozó értékek előállítását más úton is végezhetjük. Ha két ugyanott értelmezett egyértékű (f_1 és f_2) függvényünk van, akkor az ugyanahhoz a „független változó" értékekhez tartozó függvényértékeket tekinthetjük egymáshoz tartozóknak. Ezt a közös változót (s) hívjuk segédváltozónak, mivel az a szerepe, hogy segítsen a függvénykapcsolat egymáshoz tartozó értékpárjainak kiszámításában.

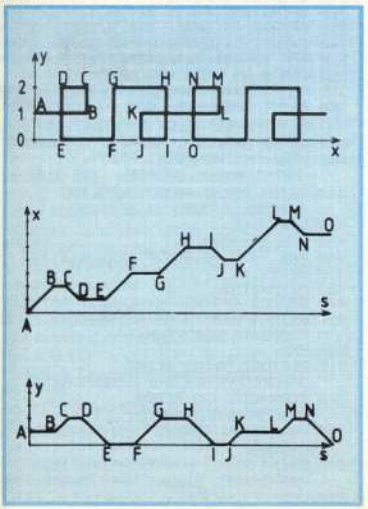
1. ábra



2. ábra



3. ábra



A segédváltozós függvénymegadási módszer tehát az, hogy nem x értékéhez számítjuk ki azt az y értéket, amely hozzá tartozik, hanem s értékéhez számítjuk ki x értékét, majd ugyanehhez az s értékhez y értékét, és az így kiszámított, ugyanahhoz az s értékhez tartozó x és y értékpár fog egymáshoz tartozni *(2. ábra)*.

Mivel ez a módszer mind a szórakoztató, mind a „komoly" számítástechnikai grafikában nélkülözhetetlenül fontos, kissé részletesebben foglalkozunk vele.

Első favágó feladatunk a „mozgáskészség" begyakorlására szolgál. Mivel leggyakrabban az előállítandó görbe ívhosszát használják segédváltozónak, a gyors tájékozódás érdekében ki kell alakítanunk azt a készséget, hogy ha egy egyenletes sebességgel „utazó" pont végighalad a görbén, biztonságosan követni tudjuk, de „legalábbis érezzük", hogy hogyan mozognak a neki megfelelő x, illetve y görbét leíró pontok.

Két feladatot adunk. Az egyikben többértékű függvények görbéi és ugyanezeknek egy-egy ívhosszúság segédváltozós előállítású változatának x és y görbéje szerepel. Ki kell válogatni az egymáshoz tartozó görbéket.

Segítségképpen egy példát mutat a *3. ábra*, amelyen az egymásnak megfelelő pontok is jelölve vannak. A betűvel jelölt ábrák mindegyikén, mindegyik tengelyen az egység egyforma hosszúságú, és egyenlő az „A" ábra görbéje legkisebb egyenesszakaszának hosszával.

A második feladat egy „if nélküli" program. Meg kell állapítani, hogy ez – ideális pontosságú műveleteket feltételezve – mire való.

A segédváltozós függvények számítástechnikai megvalósításával külön foglalkozunk majd.

A feladatokat és megoldásukat Benedikti István ellenőrizte.

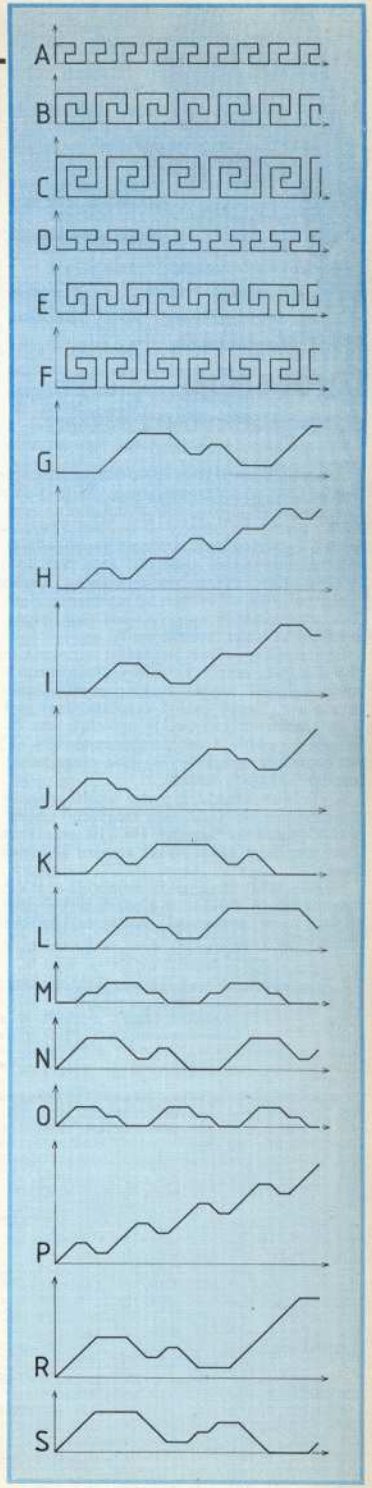
TAKÁCSY ILDIKÓ

```
10 INPUT M
20 FOR J=1 TO M
30 INPUT X(J)
40 NEXT J
50 FOR J=1 TO M-1
60 FOR I=J+1 TO M
70 LET B=(X(I)+X(J)+ABS(X(I)-X(J)))/2
80 LET A=(X(I)+X(J)-ABS(X(I)-X(J)))/2
90 LET X(J)=B
100 LET X(I)=A
110 NEXT I
120 NEXT J
130 FOR J=1 TO M
140 PRINT X(J)
150 NEXT J
160 END
```

Megoldások

1. Az egymáshoz tartozó görbék (a görbe, majd x és y az ívhossz függvényében): A–H–O; B–I–N; C–G–S; D–P–M; E–J–K; F–R–L.

2. A program rendezést végez.



HYPRA–LOAD

Gyorsabb programbetöltés

Az utóbbi hónapokban egyre több szoftver- és hardvergyártó kínál megoldást a VC 1541 és a Commodore 64 közötti adatátvitel gyorsítására. Azok a kombinált hardver- és szoftverbeavatkozások a legsikeresebbek, amelyek a soros adatátvitelt párhuzamossá alakítják, és a VC 1541 DOS vezérlő szoftvert lecserélik. Így az összes diszkettművelet sebessége öt-hatszorosára nő. Ezek a megoldások természetesen elég drágák, és nálunk nehezen hozzáférhetők.

Ezért most egy olyan megoldást mutatunk be olvasóinknak, amely kizárólag szoftvereszközökkel dolgozik, semmilyen hardvertáltoztatást nem igényel. Ennek viszont az az ára, hogy csak a programbetöltés (LOAD) művelete lesz öthatszor gyorsabb, a programkimentés (SAVE) és a diszkett programból való írása-olvasása változatlan sebességű marad.

A HYPRA–LOAD, a gyors betöltőprogram és ismertetése az NSZK-ban megjelenő „64'er" nevű Commodore magazin 1984/10. számában jelent meg, leírásunkat annak alapján állítottuk össze.

Nézzük tehát, hogyan is működik a HYPRA–LOAD. A feladat: a C64 és a VC 1541 közötti soros adatátvitel szoftvereszközökkel történő felgyorsítása.

```
0 :CLR:GOTO380
10 ************************************
11 *                                  *
12 *          HYPRA  LOAD             *
13 *          ===========             *
14 *                                  *
15 *          64'ER 84/10             *
16 *                                  *
17 ************************************
100 DATA011,008,192,007,158,051,050,051
101 DATA056,000,000,000,032,047,243,169
102 DATA226,162,248,133,167,134,168,169
103 DATA000,162,003,133,169,134,170,169
104 DATA008,032,012,237,169,111,032,185
105 DATA237,169,077,032,221,237,169,045
106 DATA032,221,237,169,087,032,221,237
107 DATA160,000,165,169,032,221,237,165
108 DATA170,032,221,237,169,030,032,221
109 DATA237,177,167,032,221,237,200,192
110 DATA030,144,246,032,254,237,024,165
111 DATA167,105,030,133,167,144,003,230
112 DATA168,024,165,169,166,170,105,030
113 DATA133,169,144,002,230,170,224,005
114 DATA144,173,201,000,144,169,169,008
115 DATA032,012,237,169,111,032,185,237
116 DATA169,077,032,221,237,169,045,032
117 DATA221,237,169,069,032,221,237,169
118 DATA139,032,221,237,169,004,032,221
119 DATA237,169,011,141,017,208,032,251
120 DATA237,234,234,076,046,240,169,011
121 DATA141,000,221,044,000,221,016,251
122 DATA169,003,141,000,221,162,005,202
123 DATA234,208,252,162,004,173,000,221
124 DATA042,042,102,176,106,102,176,234
125 DATA202,208,242,165,176,073,255,096
126 DATA032,082,239,201,255,240,248,160
127 DATA000,169,011,141,000,221,044,000
128 DATA221,016,251,169,003,141,000,221
129 DATA162,007,202,208,253,173,000,221
130 DATA042,042,102,176,106,102,176,234
131 DATA234,173,000,221,042,042,102,176
132 DATA106,102,176,234,234,173,000,221
133 DATA042,042,102,176,106,102,176,234
134 DATA234,173,000,221,042,042,102,176
135 DATA105,102,176,165,176,073,255,153
136 DATA166,251,200,208,180,096,120,169
137 DATA001,133,167,160,255,032,124,239
138 DATA192,255,240,064,162,002,165,167
139 DATA240,002,162,004,173,166,251,208
140 DATA007,238,167,251,173,167,251,044
141 DATA169,000,133,168,189,166,251,145
142 DATA174,230,174,208,002,230,175,232
143 DATA228,168,208,240,162,000,134,167
144 DATA173,166,251,208,198,169,053,133
145 DATA001,169,027,141,017,208,169,064
146 DATA133,144,024,096,169,053,133,001
147 DATA169,027,141,017,208,234,169,029
148 DATA056,096,160,000,185,066,240,153
149 DATA048,001,200,192,031,208,245,032
150 DATA210,239,076,048,001,234,008,072
151 DATA169,000,168,089,000,160,200,208
152 DATA250,201,128,240,004,169,055,133
153 DATA001,166,174,164,175,104,040,088
154 DATA096,221,016,173,240,034,173,161
155 DATA002,074,176,250,173,001,221,041
156 DATA253,141,001,221,173,001,221,041
157 DATA004,240,249,169,144,024,076,059
158 DATA239,173,161,002,041,018,240,243
159 DATA024,096,173,151,002,172,156,002
160 DATA204,155,002,240,011,041,247,141
161 DATA151,002,177,247,238,156,002,096
162 DATA009,008,141,151,002,169,000,096
163 DATA072,173,161,002,240,017,173,161
164 DATA002,041,003,208,249,169,016,141
165 DATA013,221,169,000,141,161,002,104
166 DATA096,165,000,041,006,201,002,240
167 DATA003,076,158,253,234,169,005,133
168 DATA009,162,090,134,075,162,000,169
169 DATA082,133,036,032,086,245,080,254
170 DATA184,173,001,028,197,036,240,009
171 DATA198,075,208,239,169,010,076,105
172 DATA249,080,254,184,173,001,028,149
173 DATA037,232,224,007,208,243,032,151
174 DATA244,165,022,069,023,069,024,069
175 DATA025,069,026,240,007,198,009,208
176 DATA192,076,030,244,165,024,197,006
177 DATA240,003,076,011,244,133,034,169
178 DATA006,133,049,076,060,004,165,018
179 DATA166,019,133,022,134,023,165,006
180 DATA133,024,165,007,133,025,169,000
181 DATA069,022,069,023,069,024,069,025
182 DATA133,026,032,052,249,162,090,032
183 DATA086,245,160,000,000,254,184,173
184 DATA001,028,217,036,000,240,006,202
185 DATA208,237,076,081,245,200,192,008
186 DATA208,234,032,086,245,080,254,184
187 DATA173,001,028,145,048,200,208,245
188 DATA160,186,000,254,184,173,001,028
189 DATA153,000,001,200,208,244,032,224
190 DATA248,165,056,197,071,240,003,076
191 DATA246,244,032,233,245,197,058,240
192 DATA003,076,002,245,160,000,169,085
193 DATA032,082,004,185,000,006,133,119
194 DATA044,000,024,016,251,169,016,141
195 DATA000,024,044,000,024,048,251,162
196 DATA000,138,102,119,042,042,102,119
197 DATA042,042,141,000,024,138,102,119
198 DATA042,042,102,119,042,042,141,000
199 DATA024,138,102,119,042,042,102,119
200 DATA042,042,141,000,024,138,102,119
201 DATA042,042,102,119,042,042,141,000
202 DATA024,162,002,202,208,253,169,015
203 DATA141,000,024,200,208,173,234,234
204 DATA234,234,234,234,234,173,000,028
205 DATA009,008,141,000,028,173,000,006
206 DATA208,003,076,158,253,197,024,208
207 DATA249,133,006,173,001,006,133,007
208 DATA076,101,003,133,119,044,000,024
209 DATA016,251,169,016,141,000,024,044
210 DATA000,024,048,251,162,004,169,000
211 DATA102,119,042,042,102,119,042,042
212 DATA141,000,024,202,208,240,234,234
213 DATA234,234,234,234,169,015,141,000
214 DATA024,096,096,133,000,088,165,000
215 DATA048,252,120,096,120,234,234,234
216 DATA234,234,234,165,024,141,000,006
217 DATA133,006,165,025,141,001,006,133
218 DATA007,169,004,133,120,169,226,032
219 DATA130,004,201,002,144,051,160,000
220 DATA132,120,164,120,185,219,254,240
221 DATA018,088,032,118,214,120,169,226
222 DATA032,130,004,201,002,144,026,230
223 DATA120,208,231,169,192,032,130,004
224 DATA169,226,032,130,004,201,002,144
225 DATA008,169,255,032,082,004,076,034
226 DATA235,173,000,006,240,248,197,024
227 DATA240,196,173,000,006,133,006,173
228 DATA001,006,133,007,076,160,004,234
229 DATA234,234,234,160,000,185,025,244
230 DATA153,048,001,200,192,031,208,245
231 DATA076,048,001,169,000,168,089,000
232 DATA160,200,208,250,201,128,240,007
233 DATA169,055,133,001,076,001,245,076
234 DATA048,244,160,000,177,187,201,036
235 DATA240,242,169,001,133,167,169,000
236 DATA133,144,165,167,032,012,237,169
237 DATA111,032,185,237,165,144,016,011
238 DATA230,167,165,167,201,016,208,230
239 DATA076,187,238,165,167,201,008,240
240 DATA239,160,000,185,122,244,240,006
241 DATA032,210,255,200,208,245,032,225
242 DATA255,208,251,238,234,234,234,234
243 DATA076,187,238,234,013,067,083,065
244 DATA075,032,065,032,070,076,079,080
245 DATA080,089,032,076,069,071,089,069
246 DATA078,032,066,069,075,065,080,067
247 DATA083,079,076,086,065,013,000,000
248 DATA000,000,000,000,000,120,169,055
249 DATA133,001,160,000,132,003,169,160
250 DATA133,004,177,003,145,003,230,003
251 DATA208,248,230,004,208,244,169,229
252 DATA141,214,253,169,076,141,114,254
253 DATA141,249,244,169,188,141,115,254
254 DATA169,254,141,116,254,169,009,141
255 DATA250,244,169,244,141,251,244,169
256 DATA018,133,003,169,010,133,004,169
257 DATA226,133,005,169,248,133,006,177
258 DATA003,145,005,200,208,249,230,004
259 DATA230,006,165,006,201,250,208,239
260 DATA169,016,162,008,133,003,134,004
261 DATA169,187,162,238,133,005,134,006
262 DATA177,003,145,005,200,208,249,230
263 DATA004,230,006,165,006,201,240,208
264 DATA239,185,067,013,153,154,228,200
265 DATA192,017,208,245,160,000,185,012
266 DATA012,153,009,244,200,192,148,208
267 DATA245,169,053,133,001,076,248,252
268 DATA234,234,032,032,018,072,089,080
269 DATA082,065,045,076,079,065,068,146
270 DATA032,032,032,000,000,000,000,000
271 DATA000,000,000,000,000,000,000,000
272 DATA000,000,000,000,000,000,000,000
273 DATA000,000,000,000,000,000,000,000
274 DATA000,000,000,000,000,000,000,000
275 DATA000,000,000,000,000,000,000,000
276 DATA000,000,000,000,000,000,000,000
277 DATA000,000,000,000,000,000,000,000
278 DATA000,000,000,000,000,000,000,000
279 DATA000,000,000,000,000,000,000,000
280 REM ********************************
290 REM *     KONROLLOSSZEGEK          *
300 REM ********************************
310 DATA10355,10666,10986,11346,11058
320 DATA09936,08856,08690,09460,07556
330 DATA08119,08952,09829,10579,08574
340 DATA12151,09958,08096,00000,00000
350 REM ********************************
360 REM *     TOLTOPROGRAM             *
370 REM ********************************
380 PRINT"■■HYPRA-LOAD TOLTOPROGRAM.":
    PRINT"■VOLT ELOTTE 'SAVE' ?"
390 GET A$:IFA$<>"I"ANDA$<>"N"THEN390
400 IFA$="N"THENPRINT:PRINT
    :PRINT"AKKOR GYORSAN !■■":END
410 PRINT:PRINT:PRINT"TURELEM,
    KONTROLLOSSZEG ELLENORZES !"
420 DIM S(18)
430 FOR X=1TO18
440 FOR Y=1TO80:READA:S(X)=S(X)+A
450 NEXTY,X
460 FORX=1TO18
470 READCS:IFS(X)=CSTHENNEXT:GOTO510
480 PRINT:PRINT"HIBA A";90+10*X;" -";
    99+10*X" SOROKBAN !"
500 END
510 RESTORE:PRINT:PRINT
    :PRINT"HYPRA-LOAD GENERALAS"
520 FORX=6TO1439:READA
    :POKE2049+X,A:NEXT
530 POKE45,84:POKE46,13:POKE47,84
    :POKE48,13
540 PRINT:PRINT"A HYPRA-LOAD MOST
    'SAVE'-VEL DISZKETTRE IRHATO":PRINT
550 CLR
```

Az átvitelt a C64 és a VC 1541 ROM memóriájában tárolt programok vezérlik. Ezek azonban nem használják ki maximálisan a soros busz lehetőségeit. A megfelelő buszvezérlő rutinok módosításával az átvitel sebessége megtízszerezhető. Ezt a módszert alkalmazza a HYPRA–LOAD is. Hogy a programbetöltés ideje miért nem csökken ennek megfelelően tizedrészére, annak oka a VC 1541 sebességében rejlik. Így a gyorsabb buszrutinok végül is öt-hatszoros átvitelgyorsítást eredményeznek.

A HYPRA–LOAD a BASIC területre töltődik be. RUN-nal elindítva először az egész operációs rendszert ROM-ból az azonos című RAM területre másolja, majd a szükséges rutinokat lecseréli, és az így módosított rendszert elindítja. A HYPRA–LOAD aktív állapota így a memória 1. bájtjának tartalmából olvasható le: 55 – ROM bekapcsolva; 53 – ROM kikapcsolva, HYPRA–LOAD aktív.

A már betöltött HYPRA–LOAD ennek megfelelően kikapcsolható a POKE 1,55 utasítással, és visszakapcsolható a POKE 1,53 utasítással.

Mivel a HYPRA–LOAD-dal kiegészített operációs rendszer a RAM-ban helyezkedik el, néhány problémával kell számolni. Ha olyan programot töltünk be, amely a $A000-tól kezdődő területet, vagyis az éppen aktív operációs rendszert (BASIC-et) felülírja, akkor a sikeres betöltés után a számítógép hibás működése nagyon valószínű lenne. Ilyen programok betöltésére azonban a HYPRA–LOAD fel van készítve.

Ha a $A000-lal kezdődő memóriaterület tartalma töltés közben módosul, akkor az operációs rendszerbe való visszaugrás előtt automatikus ROM-ra való visszakapcsolás történik. Több részből álló programoknál ezt rögtön észre lehet venni arról, hogy a következő fájl már a szokásos módon töltődik.

Ha egy program a $E000-tól kezdődő területet is felülírja (Kernal terület), akkor előbb vagy utóbb az éppen futó HYPRA–LOAD is sorra kerül, ami a gép azonnali leállását jelenti. Az ilyen programok HYPRA–LOAD-dal való betöltéséről általánosságban semmit nem lehet mondani, mindegyiket külön ki kell próbálni, hogy megy-e. Itt kell még megemlíteni, hogy vannak programok, amelyek szintén a HYPRA–LOAD kikapcsolását igénylik, de ezt elegendő a betöltésük után végrehajtani. Ennek felismerésében a betöltött program ismerete nélkül szintén csak a kipróbálás segíthet.

Egyetlen fájlból álló BASIC programok betöltésére a HYPRA–LOAD mindig alkalmazható, ezek a $E000-lal kezdődő memóriaterületet nem írhatják át.

Néhány megjegyzés a HYPRA–LOAD-dal kapcsolatban.

- Még egyszer meg kell említeni, hogy a HYPRA–LOAD csak a LOAD utasítás végrehajtását gyorsítja meg, a többi diszkettművelet végrehajtási ideje változatlan marad.
- A HYPRA–LOAD az operációs rendszer kazettás magnót vezérlő és RS232 működtető rutinjainak a helyére kerül, így a kazetta és az RS232 nem használható vele együtt.
- Ha a HYPRA–LOAD nincs felülírva vagy kikapcsolva, akkor SYS 64738-cal újraindítható.
- Ha a Commodore 64 Reset gombbal is fel van szerelve, ennek működtetése után a HYPRA–LOAD a POKE 40960,PEEK(40960): POKE 1,53 utasításokkal kapcsolható be ismét.
- A HYPRA–LOAD a soros buszt nagyobb sebességgel működteti, ezért az erre csatlakozó összes perifériát a betöltés végéig ki kell húzni, vagy ki kell kapcsolni. Ha ez nem történik meg, a következő figyelmeztető üzenetet kapjuk: CSAK A FLOPPY LEGYEN BEKAPCSOLVA

A RUN/STOP lenyomására ezután a betöltés elkezdődik ismételt vizsgálat nélkül!

- Ugyancsak a sebességnövelés az oka, hogy a HYPRA–LOAD a programbetöltés idejére a képernyőkijelzést kikapcsolja.
- Ha a VC 1541 nem a 8-as címen érhető el, akkor a listán látható BASIC töltőprogram 104, 114 és 239-es sorában szereplő konstanst (008) kell az új értékre lecserélni.

Ezek után hozzáláthatunk a HYPRA–LOAD BASIC töltőprogramjának begépeléséhez. Különösen gondosan ellenőrizzük a 310–340-es sorokban levő kontrollösszegek hibátlan leírását.

Az első elindítási kísérlet előtt feltétlenül mentsük ki a töltőprogramot diszkettre (SAVE "HYPRA–TOLTO",8), mert futás közben önmagát felülírja. Ezután indítsuk el RUN-nal. A töltőprogram először a DATA sorokat ellenőrzi a kontrollösszegek alapján. Ha minden hibátlannak talál, akkor a BASIC terület elejére egy gépi kód programot generál, önmaga helyére. Ezt a tulajdonképpeni HYPRA–LOAD programot rögtön diszkettre menthetjük:

```
SAVE "HYPRA–LOAD",8
```

Az így elkészült HYPRA–LOAD program a diszketten 6 blokkot foglal el. Hogy a sebességi előnyöket valóban ki tudjuk használni, ajánlatos a HYPRA–LOAD-ot minden programdiszkettre átmásolni, ne kelljen feleslegesen diszkettet cserélni.

Egy program ezután a következő utasításokkal tölthető be HYPRA–LOAD-dal:

```
LOAD "HYPRA–LOAD",8
RUN
LOAD "Program",8
RUN
```

Sok sikert kívánunk a HYPRA–LOAD használatához.

– r – k

ZX81

DATA, READ, RESTORE

A program bármelyik kezdőcímtől betölthető, ha az 55. 68. és 102. sor második és harmadik bájtját a kezdőcímnek megfelelően átírjuk.

A legfontosabb kikötés, hogy az első „értékadásnak" egy sztring dimenzionálásának kell lennie. E dimenzió értékének vagy a leghosszabb adat karakterszámánál, vagy az adatot tartalmazó sor címének karakterszámánál egygyel többnek kell lennie. Tehát

ha 20 REM ABCDEF akkor DIM A$ (6)
ha 1111 REM ABC akkor DIM A$ (5)

A DATA használata

Az adatokat REM utasításban lehet tárolni, egymástól vesszővel elválasztva. Nagyon fontos, hogy ez a REM sor nem lehet a program legelső sora. Az egymás utáni adatokat csak egy sorban lehet tárolni.

A RESTORE működése

A RESTORE-t két utasítás valósítja meg. Az elsővel a program elején dimenzionált sztringbe betöltjük annak a sornak a számát, amelyikben az éppen most beolvasandó adatokat tároljuk, a másodikkal pedig meghívjuk a gépi program első részét.

Szabvány BASIC

```
RESTORE 12
ON X RESTORE 10,20,30,40
```

ZX81 BASIC

```
LET A$ = "12."
RAND USR 16516
LET A$ = STR$(10*X)+"."
RAND USR 16516
```

A sorszám után fontos pontot is beírni. A program „R" hibajelzést ad, ha a megadott sort nem találja meg.

A READ működése

A gépi program a program elején dimenzionált sztringbe tölti be az adatot.

```
READ A$      RAND USR 16599
READ X       RAND USR 16599
             LET X = VAL A$
```

„R" hibajelzést ad, ha a program első két bájtja nulla, vagyis ha elfogyott az adat, vagy ha nem állítottuk be, hogy hol vannak tárolva az adatok.

Ha a beolvasandó adat hosszabb, mint a sztring, amit dimenzionáltunk, akkor a következő READ hatására az adatot folytatólagosan olvassa be.

Ha a beolvasandó adat rövidebb, mint a sztring, akkor a többi helyet space tölti fel.

PÉLDA

```
 10 DIM A$ (4)
 20 LET A$ = "30".
 30 REM ABCDEFGHIJKLMN,PR, O, 11 E3
 40 RAND USR 16516
 50 FOR I = 1 TO 6
 60 RAND USR 16599
 70 PRINT A$
 80 NEXT I
 90 RAND USR 16519
100 PRINT VAL A$
    RUN
```

A NYOMTATÁSI KÉP

```
 ABCD
 EFGH
 IJKL
 MN
OP
Q
 11000
```

A GÉPI PROGRAM BETÖLTÉSÉT VÉGZŐ PROGRAM

```
 10 REM 143 db space
 20 FOR I = 16514 TO 16657
 30 LET X = INT (PEEK I(16))
 40 LET Y = PEEK I – X*16
 50 SCROLL
 60 PRINT I, CHR$ (X)+CHR$(Y);
 70 INPUT A$
 80 IF A$ = "" THEN GOTO 110
 90 POKE I, CODE A$*16+CODE A$
    (2)–476
100 PRINT AT 21, 16; A$;
110 NEXT I
```

Az indítás után a *táblázat* hexadecimális számait két karakterenként kell beírni.

```
00 00 ED 5B 10 40 21 00 00 06 00 13 13 13 13 13
13 1A FE 1B 28 10 D6 1C 4F D5 54 5D 3E 09 19 3D
20 FC 09 D1 18 EA 44 4D 21 7D 40 ED 5B 0C 40 7E
FE 76 28 0F 23 7C BA 38 F6 7D BB 38 F2 3E 1A 32
00 40 C9 23 7E B8 20 ED 23 7E B9 20 E8 01 04 00
09 22 82 40 C9 ED 5B 10 40 13 13 13 13 1A 4F 13
1A 47 03 2A 82 40 7D B4 28 D3 0B 78 B1 28 19 13
7E FE 1A 28 08 FE 76 28 05 12 23 18 ED 23 AF 12
13 0B 78 B1 20 F8 18 06 7E FE 1A 20 02 23 7E FE
76 20 03 21 00 00 22 82 40 C9
```

A program leállása után a 20. sortól törölni kell a programot, és be lehet írni a nekünk megfelelőt. Célszerű a feltöltött 10. sort felvenni szalagra, hogy a későbbiekben ne kelljen újra beírni.

Ha a programot más címtől írjuk be, akkor a dőltbetűs bájtokat is ennek megfelelően meg kell változtatni.

KRIZSÁK LÁSZLÓ

Az alábbiakban néhány terméket ajánlunk a **FINOMMECHANIKAI ÉS ELEKTRONIKUS MŰSZERGYÁRTÓ SZÖVETKEZET** gyártmányaiból

PROM PROGRAMOZÓ

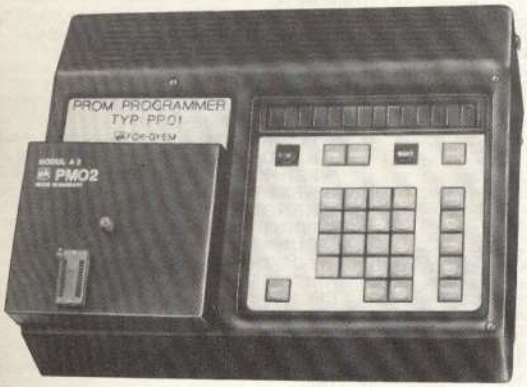
A készülék segítségével a legelterjedtebb PROM típusok programozása valósítható meg.
A készülék felépítése szerint két csoportra bontható:
- Alapkészülék egy μP vezérelt kisszámítógép, mely tartalmazza a hexadecimális billentyűzetet és gyors lyukszalagolvasó interfészt. Ezáltal szalagról is betölthető a programozandó PROM tartalma, VI, illetve INTEL formátumban. A kijelzősor és a billentyűzet segítségével a programozandó SW-ben a javítások könnyedén elvégezhetők.
- Programozó modulok a programozási HW feltételeket biztosítják a PROM-ok részére. A programozható PROM típusok a következők: P 2704, P 2708, P 2716, P 2732 és P 8755 PROM.

A felsorolt típusú PROM-ok programozásához típusonként kétfajta programozó modul választható:
- egy PROM programozása
- MASTER PROM másolása négy PROM-ba

A készülékhez tartozik a különálló UV törlő, amellyel az EPROM-ok kitörölhetők.

Műszaki adatok:

Méret:	500 × 300 × 180 mm
Tömeg:	8 kg
Hálózati feszültség:	220 V, 50 Hz
Fogyasztás:	75 A



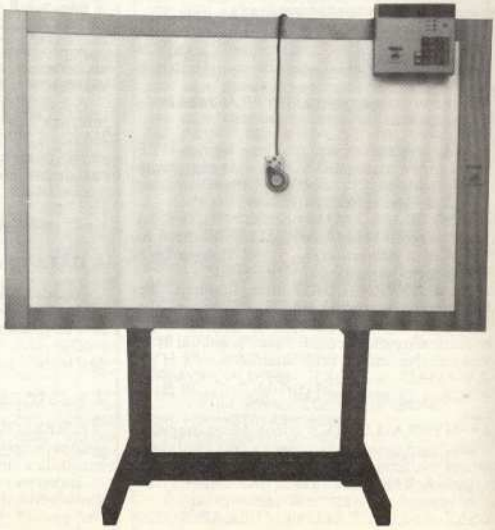
Ezek a termékek a Migértnél vagy közvetlenül a Szövetkezetnél rendelhetők meg.

INTELLIGENS RAJZDIGITALIZÁLÓ

A rajzdigitalizáló olyan feladatok megoldására alkalmazható, amelyekben grafikus vagy képi információt kell átalakítani számítástechnikai feldolgozásra alkalmas formába.

Műszaki adatok:

Munkafelület (mm)	1050 × 675
Felbontóképesség (mm)	0,1 ± 0,02
Munkafelület méretének hőfokfüggése	$2 \cdot 10^{-5}$/°C
Letapogatási sebesség	max. 3,25 m/s
Számkijelzés X és Y irányban	± 9999,9 mm vagy ± 99,995 inch
Digitalizálható anyag	max. 1 mm vastag nem mágneses anyag
Pozicionáló eszköz	szabadon mozgatható tekercs, középen hajszálkereszttel



HT–1080Z-re

MÁSOLÓPROGRAM

Az alábbi, MIXER nevű program feladata, hogy két gépi kódú programból egyet készítsen. Segítségével fantasztikus hatásokat lehet elérni.

Ha előbb egy gépi kóddal rajzot, majd pedig a kívánt játékprogramot beolvassuk MERGE utasítással, és ezt felvesszük, olyan felvételhez jutunk, amely a *beolvasás alatt* egy rajzot ír ki a képernyőre, hasonlóan a ZX–Spectrum egyes játékprogramjaihoz. Természetesen nemcsak rajzot lehet így hozzákeverni a programhoz, hanem szöveget is.

Egy-egy bájtsorozat hozzákeverésével azt is elérhetjük, hogy a program automatikusan induljon. Természetesen azt a néhány bájtot először külön rekordban fel kell venni.

Ha egy gépi kódú program úgy helyezkedik el a tárban, hogy az egyik rutin a tár elején, a másik a tár végén található, nem kell felvenni a köztük levő üres tárat is a rögzítéshez, tehát kisebb a helyfoglalás a szalagon, gyorsabb a beolvasás. Akár egy gépi „mammut" programot is készíthetünk, ahol az egyes rutinokat a főprogram külön olvassa be.

A programlista forrásprogram, amelyet az EDI nevű fordítóprogrammal kell szalagra rögzíteni. Először begépeljük a listát, vigyázva arra, hogy LOAD ne legyen benne. Ezután assembláljuk az „A" paranccsal, és a kérdésre „C" betűvel válaszolunk. Kezdőcímnek beírjuk: 06CCH, névnek pedig: MIXER. Az így nyert rekordot önmagával másoljuk, hogy a rekord tömörebb legyen. Tömören a program nem több egy fordulatnál.

A program három parancsot ismer:

LOAD – Betölt a tárba egy gépi rekordot, úgy, mintha a kurrens rekord lenne, az eddigi legelső. Az eddig betöltött rekordok elvesznek.

SAVE – A már betöltött rekordot viszi ki szalagra. Az összeset, mintha egy lenne.

MERGE – Utána tölt. Az eddig bent lévő program belépési címe elvész, az érvényes belépési pont a legutóbbié lesz.

Figyelem! Ezek a szavak csak parancs módban használhatók. Bármilyen más szerkezetben (például FOR i = 1 TO 3: SAVE: NEXT vagy LOAD: SAVE) SN hibát eredményeznek.

Aki tanulmányozta már az MCCOPY programját, azzal sok hasonlóságot fedezhet fel. Nem csoda, mert így a legegyszerűbb a program.

A MIXER nagy előnye az MCCOPY-val szemben, hogy 48 k-s kiépítésnél is használható 48 k-s programok másolására, míg az MCCOPY csak 16 k-s gépeken használható eredményesen. Ennek az az oka, hogy az egyik program maga elé veszi a rekordot, a másik maga mögé.

NAGY OTTÓ

```
                      ; MIXER
                      ;(GEPIKODU MASOLO ES KEVERO PROGRAM)
                      ; GEPTIPUS: HT-1080Z
                      ; KESZITETTE: NAGY OTTO'





                              ORG   0FFF0H     ; BOLONDITAS
FFF0 FF                       RST   56
                              ORG   400CH      ; AZ RST 28H
400C C38943                   JP    TSR        ; VEKTORA
                              ORG   40B1H      ; A BASIC VEGE
40B1 FE42                     DB    254, 66
                              ORG   4188H      ; BASIC BOVITES
4188 C32043                   JP    OLVAS      ; LOAD VEKTORA
418B C30043                   JP    MIXER ,    ; MERGE VEKTORA
                              ORG   41A0H '
41A0 C36E43                   JP    SAVE       ; SAVE VEKTORA
                              ORG   41E2H      ; AUTOMATIKUS
41E2 E9                       JP    (HL)       ; INDITAS
                              ORG   4300H
                      MIXER:
4300 2A7243                   LD    HL,(VEG+1)
4303 2B                       DEC   HL         ; ILLESZTES
4304 2B                       DEC   HL         ; AZ ELOZO
4305 2B                       DEC   HL         ; DARABHOZ
4306 0606                     LD    B,6
4308 AF                       XOR   A          ; MAGNO BEKAPCSOLAS
4309 CD1202                   CALL  0212H      ; RELE
430C CD9602                   CALL  0296H      ; KERESES
430F CD3502                   CALL  0235H      ; BYTE OLVASAS
4312 FE55                     CP    55H        ; GKPR JELE
4314 204F                     JR    NZ,ERR
                      ; A PROGRAM NEVENEK A KIIRATASA
4316 CD3502           Q1:     CALL  0235H      ; BYTE OLVASAS
4319 CD3300                   CALL  0033H      ; BYTE MEGJELENITES
431C 10F8                     DJNZ  Q1
431E 1817                     JR    Q3         ; A LOADNAL FOLYTATJA
                      OLVAS:
4320 218F43                   LD    HL,BUFFER
4323 0606                     LD    B,6
4325 AF                       XOR   A          ; A MAGNO BEKAPCSOLASA
4326 CD1202                   CALL  0212H      ; UGRAS A RELE KAPCSOLOHOZ
4329 CD9602                   CALL  0296H      ; KERESES
432C EF                       RST   28H        ; TSR RUTINHOZ
432D FE55                     CP    55H        ; GKPR JELZESE
432F 2034                     JR    NZ,ERR
4331 EF               Q2:     RST   28H
4332 CD3300                   CALL  0033H      ; VIDEORA KI
4335 10FA                     DJNZ  Q2         ; KIIRATAS
4337 CD2C02           Q3:     CALL  022CH      ; CSILLOGAS
433A EF                       RST   28H
433B FE78                     CP    78H        ; TRAILER JELE
433D 2816                     JR    Z,Q5
433F FE3C                     CP    3CH        ; REKORD JELE
4341 2022                     JR    NZ,ERR
4343 EF                       RST   28H        ; A REKORD HOSSZA
4344 47                       LD    B,A
4345 EF                       RST   28H        ; BETOLTESI CIM ALSO
4346 4F                       LD    C,A
4347 EF                       RST   28H        ; BETOLTESI CIM FELSO
4348 81                       ADD   A,C        ; ELLENORZO OSSZEG SZAMOLAS
4349 4F                       LD    C,A
434A EF                       RST   28H        ; SOR OLVASASA
434C 4F                       LD    C,A
434D 10FB                     DJNZ  Q4
434F EF                       RST   28H        ; ELLENORZO OSSZEG
4350 B9                       CP    C          ; OSSZEHASONLITASA
4351 2012                     JR    NZ,ERR
4353 18E2                     JR    Q3
4355 EF                       RST   28H        ; INDITASI CIM ALSO
4356 EF                       RST   28H        ; INDITASI CIM FELSO
4357 227243                   LD    (VEG+1),HL
435A CDF801                   CALL  01F8H      ; FELVETEL ALLJ!
435D 3E0D                     LD    A,0DH      ; MENJ A KOVETKEZO
435F CD3300                   CALL  0033H      ; SORBA!
4362 C3191A                   JP    1A19H      ; VISSZA A BASICBE
                      ERR:
4365 211E19                   LD    HL,191EH   ; A SZOVEG KEZDETE
4368 CDA728                   CALL  28A7H      ; SZOVEG KIIRASI RUTIN
436B C3CC06                   JP    06CCH      ; VISSZA A BASICBE
                      SAVE:
436E 218F43                   LD    HL,BUFFER  ; BUFFER KEZDOCIM
4371 119243           VEG:    LD    DE,BUFFER+3; BUFFER VEGCIM
4374 AF                       XOR   A          ; MAGNO BEKAPCSOLAS
4375 CD1202                   CALL  0212H
4378 CD8702                   CALL  0287H      ; HEADER IRATAS
437B 7E               Q6:     LD    A,(HL)
437C CD6402                   CALL  0264H      ; BYTE KIIRATASA MAGNORA
437F 23                       INC   HL
4380 DF                       RST   18H        ; VEGE-E A BUFFERNEK
4381 20F8                     JR    NZ,Q6      ; HA NINCSEN MEG VEGE
4383 CDF801                   CALL  01F8H      ; MAGNO ALLJ!
4386 C3191A                   JP    1A19H      ; VISSZA A BASICBE
                      ;
                      ; RST 28H VEKTOR RUTINJA.
                      TSR:
4389 CD3502                   CALL  0235H      ; BYTE OLVASAS MAGNOROL
438C 77                       LD    (HL),A     ; A BYTEOT RAKD A BUFFERBE!
438D 23                       INC   HL
438E C9                       RET
                      BUFFER:
                              ORG   0FFF7H     ; BOLONDITAS
FFF7 FF                       RST   56H
                              END
```

1 k-s ZX81-re

FOLYOSÓ

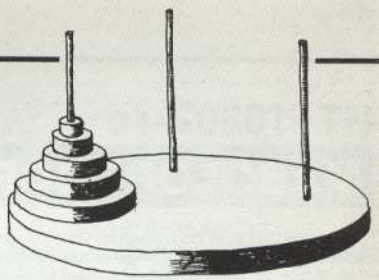
A képernyőn megjelenő falak között kell manővereznünk a „Ø" és az „1" billentyűk segítségével. A REM-be egy rövid, gépi kódú rutin kerül. POKE16514,42 : POKE16515,14 : POKE16516,64 : POKE16517,78 : POKE16518,6 : POKE16519,Ø : POKE1652Ø,2Ø1

Természetesen a ":" = NEWLINE.

NÉMETH BALÁZS

```
  1>REM 0000000
   5 LET A=NOT PI
   6 LET L=NOT PI
  10 LET A$="SHIFT 9 MAJD 12 SPACE"
  20 LET X=CODE "£"
  30 LET C=RND
  40 LET A=A+(C>=.5)-(C<.5)
  50 LET A=A+(CODE "SHIFT 9 MAJD SHIFT A"+A=NOT PI)-(CO
DE "SHIFT 9 MAJD SHIFT A"+A>LEN A$)
  60 PRINT TAB NOT PI;A$( TO 8+A);TAB 10+A;A$(8+A  TO )
  65 LET X=X+(INKEY$="0")-(INKEY$="1")
  70 PRINT AT 7,X
  80 LET N=USR VAL "16514"
  85 LET L=L+CODE "SHIFT 9 MAJD SHIFT S"
  90 PRINT AT 7,X;"*"
 100 IF N=CODE "■" THEN  PRINT AT CODE "£",CODE "<";"S
CORE:";L
 105 IF N=CODE "■" THEN  STOP
 115      SCROLL
 120 GO TO CODE "2"
```

HANOI-TORONY



A legenda szerint egy benareszi templomban réz alapzathoz van erősítve három gyémánttű. Az elsőre annak idején maga Brahma helyezett el hatvannégy aranykorongot, úgy, hogy a korongok átmérője alulról felfelé fokozatosan csökkent. Azóta papok egymást váltó nemzedékei buzgólkodnak a számukra rendelt feladaton, hogy mind a hatvannégy korongot átrakják a harmadik tűre. A korongokat egyenként kell áthelyezni annak a szabálynak a követésével, hogy egy korongra mindig csak nála kisebb korong kerülhet. Amint a papok végeznek munkájukkal, a világ mennydörgések és villámlások közepette elpusztul. (Lásd még dr. Úry László Programozási fogások című cikkét. 1984/5. szám, 14. oldal.)

Hogyan lehet megoldani a feladatot, és mi a minimális lépésszám?

Induljunk ki a legegyszerűbb esetből, amikor csak egy korong van az A tűn. Ekkor egy áthelyezéssel – A-ról C-re – készen vagyunk. Jelölje h(n) az n korong áthelyezéséhez minimálisan szükséges lépések számát, így h(1) = 1. Két korong esetén az A tűről az első (kisebb) korongot áttesszük a B tűre, majd a másodikat a C-re, végül az elsőt a B-ről a C-re, azaz h(2) = 3. .

n korong esetén rekurziót alkalmazunk. A felső n − 1 korongot áttesszük A-ról a B tűre, h(n − 1) lépésben, majd az n-ediket a C-re, végül B-ről az n − 1 korongot a C-re szintén h(n − 1) lépésben. Ebből adódik, hogy h(n) = h(n − 1) + 1 + h(n − 1). Könnyen belátható, hogy a h(n) = 2h(n − 1) + 1 rekurzív összefüggés explicit alakja (felhasználva a h(1) = 1 kezdőértéket): h(n) = 2^n-1.

Most már ismerjük a minimális lépésszámot, de nem tudjuk, milyen lépéssorozattal lehet ténylegesen áttenni a korongokat az A tűről a C tűre. Ezt a legegyszerűbben rekurzív programmal határozhatjuk meg, ugyanazt a gondolatmenetet használva, mint a lépésszám levezetésekor. (Van más megközelítés is, például az információelméletben szereplő Gray-kód egyértelműen megfeleltethető a Hanoi-torony megoldásának.)

n korong A-ról C-re való átrakását visszavezethetjük a következőkre:

(i) n − 1 korong áthelyezése A-ról B-re
(ii) 1 korong áthelyezése A-ról C-re
(iii) n − 1 korong áthelyezése B-ről C-re.

Ennek legtömörebb kódját APL nyelven lehet elkészíteni *(1. ábra)*. A Hanoi-torony nevű function bal oldali argumentuma a korongok száma, jobb oldali argumentuma pedig a tűket szimbolizáló három karakterből álló sztring. A mintapéldát négy korongra, ABC-vel jelölt tűkkel futtattuk *(2. ábra)*.

Természetesen nemcsak rekurziót támogató nyelvekben (APL, PL1, PASCAL) írható rekurzív program, de mindenesetre ezeken lényegesen egyszerűbb. Az összehasonlítás kedvéért bemutatunk egy BASIC programot is ugyanerre a feladatra *(3. ábra)*.

A rekurzív nyelvekben a fordítóprogram gondoskodik az egyes rekurziós szintek adatainak és visszatérési címeinek megőrzéséről. BASIC-ben a rekurzió szintjeinek adatellátását nekünk kell megoldanunk, legfeljebb a visszatérési címek őrzését bízhatjuk a rendszerre. Az ezeket tároló verem (gépi reprezentációtól függően) mindenesetre nem túl nagy, tehát nagyobb mélységű rekurziót igénylő feladatoknál a GOSUB utasítás nem használható. Helyette GOTO, illetve a RETURN helyett az ON V (K) GOTO C1, C2, ..., Cm utasításokat célszerű használni, ahol a V vektorban őrizzük, hogy a C1, C2, ..., Cm címek közül hányadikra kívánunk visszatérni *(4. ábra)*.

```
∇      N HANOI_TORONY HELY
[1]    →(N=0)/0
[2]    (N-1)HANOI_TORONY HELYC1 3 2⍕
[3]    N⍕". KORONGOT "⍕HELYC1⍕"
       RU'"DRO'"L "⍕HELYC3⍕" RU'"DRA !"
[4]    (N-1)HANOI_TORONY HELYC2 1 3⍕
∇
```

1. ábra

```
1. KORONGOT A RU'DRO'L B RU'DRA !
2. KORONGOT A RU'DRO'L C RU'DRA !
1. KORONGOT B RU'DRO'L C RU'DRA !
3. KORONGOT A RU'DRO'L B RU'DRA !
1. KORONGOT C RU'DRO'L A RU'DRA !
2. KORONGOT C RU'DRO'L B RU'DRA !
1. KORONGOT A RU'DRO'L B RU'DRA !
4. KORONGOT A RU'DRO'L C RU'DRA !
1. KORONGOT B RU'DRO'L C RU'DRA !
2. KORONGOT B RU'DRO'L A RU'DRA !
1. KORONGOT C RU'DRO'L A RU'DRA !
3. KORONGOT B RU'DRO'L C RU'DRA !
1. KORONGOT A RU'DRO'L B RU'DRA !
2. KORONGOT A RU'DRO'L C RU'DRA !
1. KORONGOT B RU'DRO'L C RU'DRA !


MIT OHAJTASZ ?
```

2. ábra

Rekurzív program írása nem rekurzív nyelven meglehetős óvatosságot igényel. Érdemes a feladatot először rekurzívan formalizálni (nem feltétlenül létező programnyelven), és csak azután „lefordítani".

Visszatérve a legendára, az más érdekességeket is rejt. A hatvannégy korong átrakásának lépésszáma:

$h(64) = 2^{64} - 1 = 18\,446\,744\,073\,709\,551\,615$

megegyezik azzal a búzaszemmennyiséggel, amennyit a sakkjáték feltalálója kért jutalmul. Ha a papok éjjel-nappal másodpercenként egy korongot áthelyeznének, akkor is 500 000 millió évnél tovább tartana munkájuk. Ha egy szuperszámítógéppel egymillió lépést szimulálnánk másodpercenként, akkor „csak" 500 000 évig tartana.

Túlságosan nem kell tehát izgulnunk a világ

3. ábra

```
010 REM  ****************************************
020 REM  *        H A N O I - T O R O N Y       *
030 REM  ****************************************
040 REM
050 LET A$="A" \ LET B$="B" \ LET C$="C"
060 PRINT "A KORONGOK SZA'MA ?"
070 INPUT N
080 GOSUB 130
090 STOP
100 REM
110 REM  ***        KEZDO"DIK A REKURZIO         ***
120 REM
130 IF N>0 THEN 190
140   RETURN
150  REM
160  REM *** A FELSO" N-1 KORONG A'THELYEZE'SE ***
170  REM ***          PARKOLO'PA'LYA'RA          ***
180  REM
190  LET N=N-1
200  LET W$=B$ \ LET B$=C$ \ LET C$=W$
210  GOSUB 130
220  LET N=N+1
230  LET W$=B$ \ LET B$=C$ \ LET C$=W$
240  PRINT N;". KORONGOT ";A$;" RU'DROL ";C$;" RU'DRA !"
250  REM
260  REM *** A FELSO" N-1 KORONG A'THELYEZE'SE ***
270  REM ***       A PARKOLO'PA'LYA'RO'L         ***
280  REM
290  LET N=N-1
300  LET W$=A$ \ LET A$=B$ \ LET B$=W$
310  GOSUB 130
320  LET N=N+1
330  LET W$=A$ \ LET A$=B$ \ LET B$=W$
340  RETURN
350 END
```

```
010 LET A$="A" \ LET B$="B" \ LET C$="C"
020 PRINT "A KORONGOK SZA'MA ?"
030 INPUT N
040 DIM R(10)
050 LET K=0 \ LET R(K)=1
060 GOTO 80
070 STOP
080 IF N>0 THEN 100
090   ON R(K) GOTO 70,140,220
100   LET N=N-1
110   LET W$=B$ \ LET B$=C$ \ LET C$=W$
120   LET K=K+1 \ LET R(K)=2
130   GOTO 80
140   LET K=K-1
150   LET N=N+1
160   LET W$=B$ \ LET B$=C$ \ LET C$=W$
170   PRINT N;". KORONGOT ";A$;
      " RU'DRO'L ";C$;" RU'DRA !"
180   LET N=N-1
190   LET W$=A$ \ LET A$=B$ \ LET B$=W$
200   LET K=K+1 \ LET R(K)=3
210   GOTO 80
220   LET K=K-1

■

230   LET N=N+1
240   LET W$=A$ \ LET A$=B$ \ LET B$=W$
250   ON R(K) GOTO 70,140,220
260 END
```

4. ábra

vége miatt, már csak azért sem, mert a legenda apokrif. Eduard Lucas francia matematikus – a szórakoztató logikai feladatok egyik ismert szerzője – találta ki a 19. század vége felé. Egyetlen nyitott kérdésünk maradt: a francia szerző hindu legendája hogyan kaphatta a Hanoi-torony nevet?

LOVRICS LÁSZLÓ

1 k–s ZX81-re
EGY UTASÍTÁSSAL

A képernyő 12 × 12-es tartományában négy gomb segítségével tetszőleges grafikus képet hozhatunk létre, majd ha ez megtörtént, a BREAK segítségével leállítjuk a program futását. A GOTO 100 hatására ismét előtűnik az előbb létrehozott kép. Ne használjuk a RUN 100-at, mert ez törli a változókat.

A program futtatásakor a gép először egy karaktert vár, majd ezzel a karakterrel tölti meg a hátteret. Ha space-t adunk be, akkor a képernyőn csak a grafika jelenik meg, háttér nélkül.

Ezzel a programmal magát a képet magnószalagra is felvehetjük. Ügyeljünk arra, hogy a magnószalagról bejátszott programot GOTO 100-zal indítsuk, ha a tárolt képet akarjuk látni. Memóriabővítéssel rendelkezők felhasználhatják szubrutinként.

Új grafika létrehozásánál, illetve a program kilistázásakor ajánlatos CLEAR-t használni. A Ø billentyű megnyomása után a képernyőről törölhetjük a felesleges részeket, a 9 hatására pedig folytathatjuk a rajzolást.

```
1 LETB¤=CHR¤128
5 INPUTC¤
10 DIMA¤(12,12)
20 FORI=SGNPI TO 12
30 FORJ=SGNPI TO 12
40 LETA¤(I,J)=C¤
50 NEXTJ
60 NEXTI
61 LETX=7
62 LETY=X
63 IF INKEY¤="0"THEN LETB¤=CHR¤NOTPI
64 IF INKEY¤="9"THEN LETB¤=CHR¤128
65 LETX=X+(INKEY¤="8")-(INKEY¤="5")
70 LETY=Y;(INKEY¤="6")-(INKEY¤="7")
80 PRINT ATY,X;B¤
90 LETA¤(Y,X)=B¤
95 GOTOCODE"Z"
100 FORD=SGNPITO12
110 FORO=SGNPITO12
120 PRINT ATD,O;A¤(D,O)
130 NEXTO
140 NEXTD
```

(A programlistában szereplő minden ¤ helyett $-t kell érteni.)

NÉMETH BALÁZS

1 k–s ZX81–gépre
AUTÓVERSENY – ÚJRA

A program célja, hogy a kirajzolt pályán száguldó gépkocsi kikerülje az olajfoltokat. Az ütközés nélküli lépéseket számlálva az eredmény a robbanás után jelenik meg.

A változók:
L a pontszám;
A a pálya vízszintes koordinátája;
Z az autó vízszintes koordinátája.

A program:

20–36	pályarajzolás
200–320	pályamozgás jobbra–balra
42	irányítás
44–45	ellenőrzés
48–100	olajfoltrajzolás
400–490	robbanás és befejezés

PINTÉR TAMÁS

```
  1■RAND
  2 LET L=-5
 10 LET A=20
 15 LET Z=15
 20 IF RND>.5 THEN GOTO 200
 30 GOTO 300
 31 PLOT A,0
 33 PLOT A+20,0
 35 PLOT A,1
 36 PLOT A+20,1
 37 LET L=L+1
 40 SCROLL
 41 PRINT AT 14,Z;" "
 42 LET Z=Z+((INKEY$="8")-(INKE
Y$="5") AND L>0)
 43 PRINT AT 15,Z;
 44 LET R=PEEK (PEEK 16398+(PEE
K (16399)*256))
 45 IF R=CODE "■" OR R=CODE "▐"
 OR R=CODE "▀" OR R=CODE "▄" OR
R=CODE "▜" OR R=CODE "▙" OR R=CO
DE "▟" OR R=CODE "▛" THEN GOTO 4
00
 46 PRINT "▀"
 47 IF INT (RND*10)<>9 THEN GOT
O 20
 48 LET W=(5*RND)+2
 56 PRINT AT 21,A/2+W;"▄▄"
 57 PRINT AT 20,A/2+W;"▀▀"
 59 LET R=PEEK (PEEK 16398+(PEE
K (16399)*256))
 60 IF R=CODE "■" OR R=CODE "▐"
 OR R=CODE "▀" OR R=CODE "▄" OR
R=CODE "▜" OR R=CODE "▙" OR R=CO
DE "▟" OR R=CODE "▛" THEN GOTO 4
00
 100 GOTO 20
 200 LET A=A-2
 210 IF A<0 THEN GOTO 300
 230 GOTO 31
 300 LET A=A+2
 310 IF A+20>63 THEN LET A=A-4
 320 GOTO 31
 400 FOR F=1 TO 5
 410 PLOT 2*Z+(10*RND-4),42-2*15
+(10*RND-4)
 411 PRINT "■"
 420 UNPLOT 2*Z+(10*RND-4),42-2*
15+(10*RND-4)
 421 PRINT "▀"
 430 NEXT F
 440 PRINT AT 0,11;"PONTSZAM ";L
 450 PRINT AT 10,5;"AKARSZ MEG J
ATSZANI ?"
 460 IF INKEY$="N" THEN STOP
 470 IF INKEY$<>"I" THEN GOTO 45
0
 480 CLS
 490 RUN
```

PROGRAM (ÖN)KRITIKA

Az 1984/5. számban közölt „Tényleg olyan rossz gép a ZX81?" című cikkel kapcsolatban van néhány észrevételem.

A ZX81 ROM-ja nem a 16514-es, hanem a 16509-es címen kezdődik. A 16514 a REM utáni első bájt címe.

A 16 kbájtos plusz tárnál a programok elszállása könnyen megakadályozható, ha a póttárat szigetelőszalaggal erősen a géphez rögzítjük. A módszer nálam is jól bevált.

„A gép BASIC-je" című részben a LET A=X$ példa rossz, mert formailag is hibás, helyette a LET A=VAL X$ formát használjuk. Példaprogram:

```
1  LET X$="01101220331522"
2  FOR N=1 TO 15 STEP
5  LET A=VAL X$(N TO N+2)
10 PRINT A
15 NEXT N "
```

A bíráló sorokat Békési Gábor 6. osztályos tanuló küldte be. A figyelmes olvasó ebben a kritikában is találhat egy kis hibát. Mindenesetre nagyon örülünk, hogy a fiatalok ilyen alapossággal olvassák cikkeinket. Várjuk további kritikáikat!

1984/6. számunkban a HT-gépre szóló autóverseny-program szövege és a lista nincs egyeztetve. A szöveg helyesen: A 30, 40, 50-es sorok töltik be a gépi kódú rutint. A 60-as sor POKE utasításai rendelik ezt a 280-as LINE kulcsszóhoz. A DATA sorok a 90–100-as sorba kerültek.

1984/6. számunkban az 1 k-s ZX81-gépre szóló autóverseny-program első sora helyesen: 1 PRINT AT 13, A; " ■ ■ 5 SPACE ■ ■ " A hatodik sorban "A" helyett inverz A van.

Az olvasó írja

Legutóbbi találkozásunk alkalmával egy igen jó barátom véleményét kértem a µM-ről. Azt mondta, hogy a lap nagyon tetszik neki, de felhívta a figyelmemet, hogy válaszaimat ebben a rovatban ne egyes szám első személyben, hanem az ún. udvarias többes szám első személyben fogalmazzam. Az egyes szám első személy egy kicsit nagyképű. Megpróbáltam; a mostani levelekre először csak többes szám első személyben készült el a válaszom. Bosszantó volt, azt hiszem, az udvarias többes nem megy nekem. Valószínűleg azért nem, mert önző módon a leveleket úgy olvasom, mintha nem a lapnak, hanem nekem küldték volna. Úgy érzem, mintha a levelek íróit régen ismerném, velük személyes kapcsolatban lennék. Ezért sokszor egyes szám első személyben felelek; kérem olvasóink szíves elnézését!

László Levente, 3400 Kolozsvár
str. Vlahuta nr. 7/27.

Remélem, hogy a sok levél között felfigyel e szerény kolozsvárira is. 30 éves gépészmérnök vagyok, számítástechnikával foglalkozom, hobbim a számítógép. Lapjuk olvasottsága nálunk vetekszik a HVG-éval. Ezt csak hasonló minőséggel lehet elérni. Gratulálok, és gondolom, hogy hamarosan mint havilapot fogom üdvözölni!

A lappal kapcsolatos észrevételeim a következők:

1. Mivel a kiadvány magazin és nem szaklap, legyen minél több közérdekű cikk és több, minél rövidebb szakcikk.

2. Az ember–gép kapcsolat szerintem fölösleges. Helyette inkább bővíteni kellene a programozástechnikát, beleértve a HT és Sinclairen kívül a Texas Instruments gépeket is.

3. Sokkal több szoftvert! Legyen minél több játék, mert akkor játszva meg lehet tanulni programozni. Egy külön játékprogram-melléklet kiadása szép elgondolás, de sajnos a külföldi olvasóknak gyenge vigasz. Mi hogyan juthatnánk hozzá?

4. Megnövelni a terjedelmet 75 oldalra. Meg lehet valósítani a mostani példányszám mellett is, de a reklámok növelésével. E célból a középső dupla lapon közölhetnék egy-egy mikrogép színes poszterét. Ez pénzt hozna a házhoz, meg több figyelmet a fiatalabb korosztálytól.

5. Ha elérték a 75 oldalas lapterjedelmet, akkor nyugodtan rá lehet szánni két oldalt apróhirdetésre is. Gondolom, nemcsak nekem jönne jól egy csere- és levelezőpartner.

A tanácsokat köszönjük. A HVG-vel való összehasonlítás a legnagyobb dicséretek egyike, amelyet eddig kaptunk. A TI gépekkel kapcsolatban zavarban vagyunk, úgy látom, nincs annyi ilyen gép az országban, hogy érdemes lenne rájuk programokat közölnünk. De ha ilyen igény van, nem ellenezzük.

Remélem, lapunkat hamarosan külföldön is terjeszteni fogják, akkor természetesen a mellékleteket is. A lap terjedelmének bővítése részben pénzkérdés, részben engedélyt is kell kapnunk. Van egy belső elhatározásunk is: egy kezdő lap először tartalmilag erősödjék, azután gondolhatunk a bővítésre vagy a gyakoribb megjelenésre. Végül: a lapban hirdetni már lehet, az Adok – veszek – cserélek rovatunkat erre szántuk.

Gyaraki Erzsébet, Kőröstarcsa,
Kálvin u. 3. 5622

Sajnos csak most került a kezembe lapjuk egyik száma. Azt hiszem, nincs elég propagandája. Nálunk itt vidéken csak elvétve és néha lehet látni egy-egy példányát. Talán a televízió kibírna egy kis reklámköltséget erre is, úgy mint az IPM reklámozására. Biztos, hogy nagy tábora van ennek a folyóiratnak, de lehetne még nagyobb is.

A 3. és 4. számból néhány példányt visszakaptunk a Postától. Ha valaki nem kapta volna meg, „amíg a készlet tart", elküldjük. Ami a reklámozást illeti, azt mértéktartással tesszük. Szerencsénkre a lapot egyre többen szeretik és vásárolják.

Nagy Mihály, Székesfehérvár,
Velinszki u. 11. 8000

Azzal a kérdéssel fordulok Önökhöz, hogy hol találok MICRO baráti kört, valamint mikor és hol vásárolhatom meg azt a tankönyvet, amelyet a tévésorozathoz készítettek.

Érdekelne továbbá minden Önök által kibocsátott szakirás. Ezek megjelenéséről és a hozzájutás módjáról szíveskedjenek értesíteni.

Több társammal szeretnék beszerezni egy-egy Primo személyi számítógépet, mivel a kezdeti gyakorláshoz számunkra ez elegendő volna. Tehát érdekelne, hogy hol és milyen teljesítményűt rendelhetnénk meg, mennyibe kerülne és mi a megrendelés módja.

1. A µklubok névsorát folyamatosan közöljük, de minden megyei Neumann szervezetünknél is van µklub, a Fejér megyeiről például a Technika Házában kaphat felvilágosítást; a könyvet bármelyik könyvesboltban megveheti (bizonyára, mire lapunk megjelenik, már meg is vette).

2. Legyen az NJSZT tagja, akkor automatikusan megkapja értesítőnket, amelyben nemcsak a különféle eseményekről szerezhet tudomást, hanem a megjelenő szakanyagokról is.

3. A Primo gépről az Elektromodul (Bp. XIII., Visegrádi u. 47/a–b) ad felvilágosítást.

Petuska Tibor, Baja,
Petőfi Sándor u. 15/a. 6500

Első gimnazista vagyok, és szeretnék benevezni az iskolámban hirdetett számítástechnikai versenyre. Kész is van a programom, de hiányzik belőle egy olyan gépi kódú programrész, amely meggátolná a kilistázását, illetve a program megállítását. Kérem, legyenek szívesek megírni nekem ezt a gépi kódú programot. Megjelenése óta olvasom a µM-t, de eddig nem láttam benne ilyen programot.

Először kapunk ilyen kérést. Sürgősen továbbítottam a szakembereknek.

Eszes Zoltán, Pétervására,
Vörös Hadsereg út 85. 3250

Az 1984. december 1-i Népszabadságban olvastam, hogy Primo klubokat szervez a társaság. Nem hiszem, hogy mi esélyesek lennénk egy gépre, de legfeljebb tájékozódom ez ügyben.

A szakmunkásképző intézetben, ahol tanár vagyok, megszerveztem a HT–2080Z gépre alapozva egy számítástechnikai szakkört. Mivel mezőgazdasági gépész üzemmérnök vagyok, nem számítástechnikus, csak az eredendő kíváncsiságom késztetett arra, hogy mind mélyebbre ássak a bitek között. A községben is egyre többen érdeklődnek a számítástechnika iránt, és ez a cikk adta az ötletet a községi klub alakításához.

Tehát ha úgy érzik, hogy érdemes lenne egy gépet ilyen kezdő körbe is kiadni, nagyon szívesen fogadnánk.

Válaszomat már elküldtem. A sajtóban megjelent cikket nem tudom, ki adta le, de sajnos nem igaz, hogy minden µklubnak gépet tudunk adni. A megyei Neumann szervezetekbe küldtünk gépeket, ezek meg is érkezhettek. Elég távoli álmom, hogy minden településen legyen µklub, néhány számítógéppel, de remélem, ez az álom is megvalósul egyszer.

Rikk János, Nyíregyháza,
Damjanich u. 11.

Commodore VC–20-as gépem van. Erről magyar nyelvű írásos anyagot eddig nem találtam. Ezért segítséget kérek Öntől. A gépem tárát szeretném kibővíteni, és ehhez rajzot, az alkatrészek felsorolását és az alkatrészek beszerzéséhez címeket kérnék. Ha nem tud segíteni, legyen szíves megírni, hova fordulhatok a problémámmal.

Még egy kérésem lenne: ha tudna nekem játékprogramot küldeni, szívesen venném.

Sajnálom, egyelőre nem tudunk segíteni, noha egyre nagyobb az igény, miután elég sokan vásárolnak VC–20-as számítógépet. Ha kapunk, közölni fogunk programokat erre a gépre is, de nem biztatom. Remélem, levelére valaki válaszolni fog, és talán segít is.

Baky Miklós, Nagykőrös,
Kecskeméti u. 5/c. 2750

Gera Jánosnak ebben a rovatban megjelent írása bátoritott fel a mellékelt program megírására. Egyben szeretném bejelenteni, hogy iskolánk vezetősége engedélyezte a µklub megalakítását. Egyelőre csütörtöki napokon 16–18 óra között tudunk gépidőt és tanácsadást biztosítani három HT gépünkön.

Ezért szeretem ezt a rovatot, és azt is szeretném, ha idők folyamán a rovat az olvasók fórumává válna. Már hallottam, hogy a megjelentetett levéltöredékek alapján levelezés, együttműködés, sőt barátság alakult ki azonos érdeklődésű számítógépes amatőrök között. Ez a célunk.

József Attila Gimnázium, Makó,
Szabadság tér 6. 6900

Tanárok és tanulók örömmel olvastuk a µM 1984/3. számában a számítástechnikai népoktatásra szóló felhívást. Ennek érdekében µklubot alakítunk, lévén iskolánkban 3 db HT–1080Z iskolaszámítógép + megfelelő szaktanterem + lelkes gárda, amely a működés feltételeit szakmailag biztosítja.

Remélem, más iskolák is követik a nagykőrösi és makói példát, µklubokat alakítanak, és megengedik, hogy a diákok mellett az érdeklődő felnőttek, például a szülők is tanácsot kapjanak és géphez jussanak.

Kovács István, Eger,

Kodály Zoltán u. 2. 3300

Szeretnék érdeklődni az iránt, hogy hol van olyan iskola vagy intézmény, ahol levelező tagozaton felsőfokú programozói végzettséget lehet szerezni. Jelenleg az egri HÁÉV operátora vagyok, és szeretném magamat továbbképezni.

A magazin szerkesztésében azt hiányolom, hogy nem közlik az egyes személyi számítógépek árát és beszerzési lehetőségét.

Pontos tájékoztatást a SZAMALK Oktatási osztálya (Bp. Szakasits Árpád út 68. 1115) ad, illetve a felsőoktatási intézményekről szóló, évenként újra megjelenő tájékoztatóból szerezhet.

Fehér Csaba, Keszthely,

Fürst Sándor u. 10. 8360

Levelét megkaptam, és örömmel közlöm, hogy problémám már megoldódott. Időközben ugyanis forgalomba került a Primo személyi számítógép, és sikerült egyet beszereznem. Most már csak az érdekelne, hogy hol jelentkezhetnék a Primo klubba, amelyről mindössze annyit tudok, hogy megalakult.

A legközelebbi µklub, amelyről tudok, Veszprémben van, az NJSZT megyei szervezeténél. A titkár: Csermely Zoltán, MÁFKI, Veszprém, Pf. 92. 8201.

Szedmer József, Esztergom,

Mártírok útja 17. 2500

Nagyon furcsának tartom, hogy már több, mint egy éve él a µM, és még mindig nem tudták megoldani, hogy az előfizetők az újságárusokkal egyidőben megkapják a folyóiratot. Így egyáltalán nem látom értelmét annak, hogy a jövőben előfizessem. Azt sem tudom, miért nem teljesítik a beígért havi megjelenést. Lehet, hogy ezért nem Önök a felelősek, de akkor is felháborítónak tartom. Az előfizetési díjat természetesen be kell fizetni, a kiegészítésről nem is beszélve. Ezzel azonban olyan szerződés jött létre közöttünk, amiben Önök a havi rendszeres teljesítést vállalják. Ennek azonban eddig nem tettek eleget. Már csak azt szeretném tudni, hogy a befizetett előfizetési díj összege hogyan fog realizálódni a megjelent darabszámokban.

A lap tartalmával kapcsolatban csak annyit, hogy a számítógépes nyelv ezúton történő megtanulásáról már lemondtam. A gépépítésről azonban nem, amiben az Önök minden újabb számában megjelenő hitegetésük is „támogat", de eddig semmi tényleges, érdemleges segítséget nem nyújtottak. Mindenki ilyen titokként kezeli az épített gépét? Akkor közöljék egy KIT alapján a megepítés módját, és ne azt írják minden fentiekre utaló cikkük végére, hogy az érdeklődők forduljanak az íróhoz. Kérem, tegyenek már eleget maradéktalanul saját ígéretüknek, és segítsenek sokunkon.

Nincs kifogásom az erős hangvétel ellen, ha annak használata jogos. Valamennyi kérdésére válaszolok.

1. A terjesztés, amióta a µM hivatalosan is lappá vált, a Posta monopóliuma, ott kell reklamálnia, de

2. nem ígértünk havi megjelenést, hanem évi 6 számot. Igaz, az 1984. évi utolsó 1985 januárjában jelent meg. Hogy miért, olvassa el a 6. szám szerkesztőségi cikkét.

3. A számítógépes nyelvnek az újságból való megtanulása – ábránd. Ezt nem is vállaltuk, erre ott vannak a szakkönyvek és talán segít a TV BASIC. Kár, hogy számított erre.

4. A számítógép-építő sorozathoz nem ígértünk alkatrészt vagy KIT-et. Azt sem mondtuk, hogy egy teljesen kezdő az újságcikkekből mindent megért. Dr. Simonyi Endrét arra kértük, hogy írja le, hogy építette meg a gépét, adjon rajzokat, ez van a cikkben. Az író, mert szeretné, ha követnék, segít mindenkinek, és tanácsot ad annak, aki hozzá fordul.

Lovranits József, Székesfehérvár,

Géza u. 63. 8000

Levelemhez mellékelek egy kipróbált, működő BASIC programot HT–1080Z számítógépre. A program a memóriában valahol elhelyezkedő gépi kódú programot írja át DATA utasításrozattá.

Egyébként két éve foglalkozom a HT–1080Z számítógéppel. Diákjaimmal az első számtól kezdve lelkes olvasói vagyunk a µM-nak.

Maradjanak is, a programot köszönjük.

Nagy Tibor, Budapest,

Irinyi József u. 42. 1117

Közölni szeretném, hogy én is indulni kívánok a sakkprogramozási pályázaton, valószínűleg egy Commodore 64-es gépre írt programmal. Szeretném tudni, hogy a beküldött programmal továbbra is a szerző rendelkezhet-e (versenyen való indítási jog, eladási jog fenntartása, további felhasználásról való döntés jogának megtartása stb.), vagy a szervezők kapják meg ezt a jogot teljesen, esetleg részben? Tudják-e garantálni a program védettségét?

Egyébként a témában eddig megjelent cikkek eléggé meseszerűek, gyakorlatilag megfoghatatlanok. Gondolom, az érdeklődés felkeltése volt a cél. Lesz-e konkrétabb szakmai segítség? Található-e valahol ilyen jellegű szakirodalom, főleg az „értelmes lépés" kiválasztásáról?

A pályázatra nevezett program tulajdonjoga és minden más joga a szerzőé marad. A program védettségét nem tudjuk garantálni, illetve csak annyit, hogy mi nem használjuk fel és nem másoljuk le. Ami a cikkeket illeti, kevesen értenek a sakkprogramozáshoz, de talán Mérő László sorozata segít.

Németh Csaba, Sopron,

Várfal u. 2. 9400

Örömmel láttam újságjukban, hogy TI–99/4a tulajdonosokat keresnek TI klubba. Én is TI-tulajdonos vagyok, és szeretnék klubtag lenni. K.b. 70 programom van (játék, nyelvi stb.) és rengeteg lista. Kazettát is szívesen küldök cserélni, de más is küldhet kazettát, hogy vegyek rá programot.

Rajta, TI–99 tulajdonosok – válaszoljanak!

Müller Szilvia, Budapest,

Szigony u. 6. 1083

Nyáron kaptam egy Commodore VC–20-as számítógépet, és hozzá egy tájékoztató könyvet, sajnos német nyelven. Állítólag valamelyik gmk lefordította. Szeretném megtudni, hogy kik fordították le, és hogy hol lehetne megvenni ezt a könyvet.

Sajnos nem sikerült a kiadót megtalálnom. Lehet, hogy tud valaki segíteni?

Hipszki József, Tapolca,

Néphadsereg út 7. 8303

Az 1984/3. számban Veres József gimnáziumi tanár kérését olvasva, én is szeretném a „Zacskós számítógép" akcióra előjegyeztetni magam.

A magazinnal kapcsolatban észrevételem, hogy igen rendszertelenül és késve jelenik meg, így a megjelölt határidők vagy lejártak, vagy nagyon rövidek. Ezért sokan lemaradnak a különböző felhívásokról.

Ha lesz KIT számítógép, értesítjük. Reméljük, 1985-ben a lap rendszeresen jelenik meg, minden páros hónap közepén.

Vörös Csaba és Hirt Albert, Kisgyalán,

Kossuth u. 37. 7279

Rovataik közül a µKlubot és a Tanfolyamot tartjuk a legjobban sikerültnek, abból is a számítógép-építő tanfolyam indítását. A Sakkprogramozás című fejezetet viszont egy kicsit elvontnak tartjuk. Nagyon örülünk a lapban megjelenő személyiszámítógép-reklámoknak, rövid ismertetésüknek, de hiányoljuk a vásárló szemszögéből legfontosabbat, a gépek árát!

Köszönjük.

Sebestyén Ferenc, Esztergom,

Kölcsey Ferenc u. 10/a. 2500

Katonaidőmet töltöm, novemberben szerelek le. Tavaly vettem egy 16 k–s ZX-Spectrum gépet, ami eddig jól működött, de most elromlott. Már régebben foglalkozom ezzel a géppel, de sajnos nem tudom megállapitani a hiba okát. Kérem, segítsenek.

Megpróbáltam segíteni, de eredmény nélkül. Nem tudják, hogy a ZX–Spectrumot javítják-e valahol.

A levelek többségére levélben is válaszolok; miután sok levelet kapunk, majdnem mindig késve. Olvasóink eddigi megértését köszönöm.

KOVÁCS GYŐZŐ

Kedves Olvasónk!

Szíves tájékoztatására az alábbiakban közöljük a Lapkiadó Vállalat azon budapesti és vidéki üzleteinek névsorát, ahol lapunkat, a Mikroszámítógép Magazint megvásárolhatja.

BUDAPEST

VII., Almássy tér 11.
VII., Dohány u. 82.
VII., Dohány u. 84.
IX., Üllői út 119.
XII., Böszörményi út 3/a.
XIII., Rájk L. u. 24.
XIV., Thököly út 21.

VIDÉK

Bükkfürdő pavilonsor
Debrecen, Simonfi u. 1.
Győr, Árpád út 63.
Kőszeg, Jurisics tér 10.
Sopron, Orsolya tér Lábas ház
Szeged, Bajcsy Zs. u. 19.
Szombathely, Berzsenyi D. tér 3.
Pécs, Jókai u. 25.

LÉPÉSRŐL LÉPÉSRE

Az eddigiekben megismerkedtünk az alfa–béta algoritmussal, ami a jelenlegi legerősebb sakkprogramok „vázát" alkotja. Eközben végig fekete dobozként kezeltük a kiértékelő függvényt. Ez az a függvény, amely minden sakkálláshoz egy számot rendel, úgy, hogy minél kedvezőbb számunkra az állás, annál nagyobb a hozzárendelt szám. Az alfa–béta algoritmus azt a lépést határozza meg, amely az általunk vizsgált részfában az ellenfél lehető legjobb játéka esetén is lehetővé teszi számunkra, hogy az egyáltalán elérhető legnagyobb kiértékelőfüggvény-értéket el is érjük.

Úgy tűnik tehát, hogy a sakkról való ismereteinket ezen a ponton kell a programba beépítenünk: az optimális az lenne, ha a kiértékelő függvény szimulálja egy nagymester állásértékelését. Ezt a célt azonban nemhogy elérni, de még csak közelíteni sem sikerült, és hogy miért nem, annak elemzésére a későbbi folytatásokban még igen részletesen kitérünk.

Egyelőre csak annyit jegyezzünk meg, hogy az a mód, ahogy a minimax eljárás a kiértékelő függvényt kezeli, túlságosan statikus: az adott állásban meglevő összes fantáziát, lehetséges eszmét, veszélyt, előnyt és hátrányt egyetlenegy számba kell sűríteni. Egy sakkozó gondolkodása sohasem ilyen „egydimenziós".

Ha egyetlen számmal kell kifejeznünk az állás értékét, nemigen tehetünk mást, mint hogy megnézzük az állás különféle jellemzőit, és ezek mindegyikéhez hozzárendelünk egy-egy pontértéket: nagy pozitív számot akkor, ha a szóban forgó jellemzőt nagyon előnyösnek ítéljük meg, nagy negatívot akkor, ha nagyon hátrányosnak, és valamilyen közbülső számot a közbülső esetekben. A végén pedig ezeket az értékeket összeadjuk.

Amit a legegyszerűbb így mérni, az az anyagi egyensúly. A sakkozók már a kezdet kezdetén általában megtanulják a figurák értékét, például hogy a gyalog 1 pont, a futó és a huszár 3, a bástya 5 és a vezér 9. Ezt azután hamarosan elfelejtik, enélkül is tudják a figurák értékét. Egy kicsit is haladottabb sakkozó sosem mond olyat, hogy „egy ponttal vezetek", ha például bástya és két gyalogja van az ellenfél futója és huszárja ellen. Más dimenziókban fejezi ki állásának jóságát vagy gyengeségét.

A kiértékelő függvény azonban csak ilyesmit tud mondani. A figurák értékelése egész enyhén változik programról programra; van, ahol a futó 3,1 pont és a huszár csak 2,9; van, ahol fordítva – ez a programozó megítélésén múlik. Van program, amely finomabban disztingvál, bizonyos típusú helyzetekben a futót, másokban a huszárt értékeli csekéllyel magasabbra.

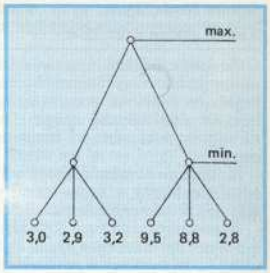
A kiértékelő függvényben azonban nemcsak az anyagi helyzetnek kell kifejezésre jutnia, hanem az állás egyéb jellegzetességeinek is. Így közös nevezőre kell hozni mondjuk egy nyílt vonal előnyét egy kettős gyalog hátrányával és az egészet egy gyalog értékével. Ezért a legtöbb program az utóbbit egyszerűen egységnek választja. Ez az oka annak, hogy némelyik program, amely játék közben kiírja azt is, hogyan értékeli állását, úgy fogalmaz, hogy „van 0,327 gyalog előnyöm". Ez a mondat azt jelenti, hogy a legutóbbi lépésnél a minimax algoritmus a kiinduló csúcshoz ezt az értéket rendelte.

Minél bonyolultabb a kiértékelő függvény, minél több finom szempontot vesz figyelembe, annál lassabban fut le, így annál kisebb fát tud a program adott idő alatt megvizsgálni. Ez az egyik oka annak, hogy még a legerősebb sakkprogramok kiértékelő függvényei is meglepően egyszerűek. A másik ok az, hogy minél több szempontot veszünk figyelembe, annál nagyobb eséllyel fognak egymásnak ellentmondó eredményt adni, és fogják így egymás hatását kioltani. A tapasztalat az, hogy hiába próbálták a kiértékelő függvényt sokkal árnyaltabbá tenni, a program játékereje ettől nem javult, sőt, volt úgy, hogy romlott.

Ken Thompson, az 1980. évi világbajnok BELLE program szerzője (akit számítástechnikai berkekben mint az UNIX rendszer és a C nyelv alkotóját is ismernek), leírta, hogy a BELLE kiértékelő függvénye az állás mely sajátosságait veszi figyelembe. Az egyes jellemzőkhöz rendelt pontos számértékeket ő sem közölte, de a vizsgált jellemzők mibenlétét igen. Ezek a következők:

1. Az anyagi egyensúly
2. Mennyire mozgékonyak a felek tisztjei (egyszerűen leszámolja, hogy a tisztek hány mezőre léphetnek)
3. Van-e izolált gyalog
4. Kettős gyalogok
5. Szabad gyalogok
6. Nyílt vonalak és átlók
7. A király biztonsága (sáncolt, ill. sáncolhat-e a király, mennyire épek a sáncállás gyalogjai)

Meghökkentően kevés. Ennyit egy kezdő sakkozó is tud a sakkról, sőt sokkal többet. Márpedig a BELLE programba, amely hivatalosan elnyerte az USA mesteri címét, ennél több sakktudás nincs beépítve. Ebből látszik, hogy a minimax algoritmus minden merevsége, egydimenzióssága ellenére nagyon is jól kihasználja a gép rendkívül gyors működési sebességéből eredő mennyiségi lehetőségeket. A BELLE program, amelyben a kiértékelő függvény számítását külön célhardver végzi, a rendelkezésére álló átlagosan 3 perc gondolkodási idő alatt húszmillió csúcsból álló fát képes kiértékelni. Egy nagymester ennek milliomodrészét sem szokta megnézni.



Ebben az állásban a minimax algoritmus a bal oldali lépést választja. De biztos, hogy az a célszerű?

Azt mondtuk, hogy a minimax algoritmus merev. Ennek illusztrálására nézzük meg az *ábrán* látható minifát. A minimax algoritmus a bal oldali ágat választja, hiszen ott az ellenfél legjobb lépése után is elér 2,9-es értéket. Könnyen lehet, hogy az ember szívesebben lépné a jobb oldali lépést, hiszen ott ha az ellenfél megtalálja a legjobb ellenlépést, akkor ugyan csak 2,8-at érünk el, de van esély sokkal többre is, míg a másik lépésnél nincs. Az ember döntése valószínűleg attól függne, mennyire tartja lehetségesnek, hogy az ellenfél nem találja meg a helyes lépést. Lehet, hogy már akkor is megéri a dolog, ha ugyan megtalálja, de túl sok időt tölt el vele. Ráadásul lehet, hogy a kiértékelő függvény bizonytalansága jóval 0,1 pont fölött van.

Nagyon ígéretesnek látszik a minimax algoritmust úgy továbbfejleszteni, hogy ilyen esetekben kicsit „toleránsabb" legyen, ne feltétlenül az abszolút optimumot keresse. Az ilyen próbálkozások eddig nem vezettek sikerre: bár az ilyen programok léptek néhány váratlan jó, „éles" lépést, de gyakran egészen durva hibákhoz vezettek. Jobb esélynek látszik a vizsgált fa meghatározását finomítani a **B** típusú stratégiák fejlesztésével.

Sakkprogramozási pályázatunk résztvevőinek!

Figyelemmel a pályázat iránt megnyilvánuló széles körű érdeklődésre és arra, hogy többen nemrég jelentették be részvételi szándékukat, a beküldési határidőt június 30-ról **1985. december 31-ig** meghosszabbítjuk. A pályázni szándékozóknak a Magyar Sakk Szövetség Számítógép Bizottsága és a Neumann János Számítógéptudományi Társaság Sakkprogramozási Munkabizottsága szívesen rendelkezésre bocsátja David Levy legújabb, sakkprogramozási ismereteket nyújtó tanulmányának magyar nyelvű fordítását, továbbá A. T. Marsland professzornak tavaly augusztusban az NJSZT-ben tartott előadása változatát, és rövidebb időre, esetleges másolásra egyéb angol nyelvű szakirodalmat is. Az ilyen kérelmeket szerkesztőségünk továbbítja.

A pályázatok a szerzők tulajdonát képezik, és azokkal szabadon rendelkeznek. Titkosságukat a szervezők és a bíráló bizottság garantálja.

Kérjük, hogy akik a pályázaton részt kívánnak venni, közöljék szerkesztőségünkkel, és írják meg azt is, munkájukkal hol tartanak, várhatóan mikorra fejezik be.

Ezen az úton ugyanis a minimax algoritmus megtartásával is bele lehet további sakktudást csempészni a programba. Előírhatunk számtalan olyan elvet, hogy bizonyos típusú lépéseket mindig vegyünk fel a vizsgált fába. Így például ha az ellenfél még nem sáncolt, és lehetőség van futóval f7-en (illetve f2-n) gyalogot ütni, akkor ez a lépés mindig szerepeljen a megvizsgáltak között. Ez többnyire fölösleges próbálkozás, de mivel sok típuskombináció kezdődik így, időnként a program pusztán ettől a kiegészítéstől meg fog találni olyan kombinációkat, amelyeket különben a **B** típusú stratégia természeténél fogva nem venne észre.

A minimax algoritmus ilyenfajta kiterjesztéseivel nagyon sok típuslépésre fel lehet készíteni a programot, nagymértékben növelni lehet a játékerejét. Vegyük azonban észre, hogy a program ettől még nem sakkozik mélyebben, semmit sem ért az állásból, a csattanós lépések folytatását nem azért találja meg, mert a szellemes lépésnél már volt fogalma a folytatásról, hanem a továbbiakat már a „vak" minimax algoritmus is tudja, ha az egész kombináció belefér a vizsgált fába.

Ez az észrevétel rávezet egy másik útra is, ahogyan saját sakktudásunkat beépíthetjük a sakkprogramba. A minimax algoritmus ugyanis nagyon erős a viszonylag rövid, konkrét variációk számolásában, de gyengéje az, hogy semennyire sem tud hosszabb távra előre tervezni, távlati koncepciót építeni. *Paradox módon a gép éppen abban erős, amit az ember intuitív, kombinatív játéknak tekint, és abban gyenge, amit az ember gépies, lélektelen játéknak észlel.*

Ha ezt felismertük, megtehetjük, hogy úgy alakítjuk ki a gép megnyitási repertoárját, hogy az lehetőleg minél élesebb játékra vezessen. Így van ugyanis a legtöbb esély arra, hogy viszonylag rövid, még a gép által belátható távon előnyt lehessen szerezni.

A megnyitási változatokat a legtöbb programba egy az egyben beprogramozzák, és itt nagy szükség van a szakszerű, de a gép stílusát értő szelekcióra: a minimax algoritmusra alapított programok ugyanis teljesen „elfogulatlanul" játszanak, nem befolyásolják őket a divatos eszmék, csakis kiértékelő függvényüket maximalizálják. Így általában nem a megszokott, elfogadott megnyitások felelnek meg a legjobban a gép stílusának, hanem azok, amelyekben a kiértékelő függvény általában jól működik. Például ha a kiértékelő függvény erősen negatívra értékeli a kettősgyalogot, akkor kerülni kell azokat a megnyitásokat, ahol a kettősgyalog erősnek bizonyul.

MÉRŐ LÁSZLÓ

(Folytatjuk)

A kabalapályázatra érkezett

13. *Szécsi Hedvig, Budapest*

A kabalapályázatra érkezett

14. *Németh Szabolcs, Keszthely*

100 FELADATOT 1 PROGRAMMAL OLDHAT MEG hazai és külföldi mikroszámítógépen

Magyarországon az első általánosan használható adatkezelő és döntéselőkészítő programrendszer, melynek használatát bárki 1 órán belül elsajátíthatja.

EZÉRT SZÍVES FIGYELMÜKBE AJÁNLJUK A CSB–805 PROGRAMOT.

A szakemberek már a Gyulán megrendezett Szervezési Akadémia, a nagyközönség pedig az Orgtechnik óta ismeri és használja. A közönségsiker alapján a VT 20/A és VT 20/IV számítógépekre írt programot ma már

- Commodore 64 gépen is lehet futtatni;
- a Műszertechnika által gyártott TZ–80 és Transmic 8 professzionális számítógépeken már merevlemezes változatban is futtatható, a Multi Centerre történő átalakítás pedig folyamatban van;
- Apple II számítógépen ugyancsak felhasználható;
- akinek MO8X-en nincs adatkezelő rendszere, annak is figyelmébe ajánljuk az alábbi rövid programismertetőt.

A CSB–805 programrendszer feladata az, hogy hatékony segítséget nyújtson az információk rendszerezéséhez és a bennük való gyors eligazodáshoz.
A CSB–805 rendszer lehetőséget nyújt pl. különböző tevékenységek, folyamatok, pénzügyi akciók figyelésére, követésére, személyi és állóeszköz-nyilvántartásra, gazdasági tervek és döntések előkészítésére, statisztikák készítésére, hálótervek, külső kapcsolatok, levelezések nyilvántartására, akár 100 különböző feladat elvégzésére.

Az adatállományt kezelő rendszer két feladatot lát el:
- a nyilvántartani kívánt adatokat rögzíti és tárolja,
- a tárolt adatokat kívánt szempontok szerint előkeresi.

Ez tükröződik a program felépítésénél is: használatánál két fő ágon lehet elindulni. Minden olyan utasítás, mely a tárolt adatokon valamilyen változást eredményez, a KARBANTARTÁS funkción keresztül érhető el. Az információk előkeresése, csoportosítása a LISTÁZÁS funkció segítségével lehetséges.

A PROGRAM FŐ FUNKCIÓI

A KARBANTARTÁS funkcióval lehet
- új adatfájlokat létrehozni,
- törölni adatfájlokat,
- módosítani adatfájlokat,
- több terminál esetén a terminálok kiszolgálását elindítani.

A LISTÁZÁS funkcióval lehet
- az adatfájlok tartalmát megjeleníteni,
- az adatokat tetszőleges szempontok szerint csoportosítani,
- az információkat a sornyomtatón papírra rögzíteni.

A KARBANTARTÁS és a LISTÁZÁS funkciók mellett a CSB–805 programrendszer több olyan lehetőséget nyújt, amely megkönnyíti a program és az adatállomány kezelését, illetve kiszélesíti a felhasználás körét. Ilyenek pl. az alábbi funkciók:
– Alfáj! – kijelölés (tetszőleges);
– Sémaszerkezet megjelenítés;
– Lemezegység-kiválasztás;
– Rendezés.

A CSB–805 rendszerben a kezelendő információhalmazt – adatállományt – a felhasználói igényeknek megfelelően kell létrehozni. Ez az adatállomány a működés folyamán tetszőlegesen módosítható, fejleszthető.
Az adatállomány egymástól független adathalmazokból – fájlokból – épül fel. Az azonos típusú információkat egy-egy fájl tartalmazza. Az adathalmazok által alkotott adatállomány megfelelő számítógépkonfiguráció esetén egyidejűleg több munkaállomásról – terminálról – is elérhető.
A terminálokról az adatállomány információi megadott szempontok szerint csoportosíthatók, rendezhetők, lekérdezhetők. Szükség esetén az egyes terminálokról kezdeményezve a rendszerben lévő printeren lista készíthető a kigyűjtött információkról.

AZ ALKOTÓ IFJÚSÁG EGYESÜLÉS MEGRENDELŐI SZÍVES FIGYELMÉBE AJÁNLJA AZ ALÁBBI PROGRAMOKAT:

Commodore 64 személyi számítógépre:

- Rendelésnyilvántartás
- Főkönyvi kivonat készítés
- Szerződés nyilvántartás
- Mérlegellenőrző program
- Célrészjegy nyilvántartás ÁFÉSZ-ek számára
- Üzemanyagelszámolás
- Pénzügyi bevételeket és kiadásokat gyüjtő program
- Költségfelosztó programcsomag
- Késedelmi kamat számítás
- Háztáji tejelszámolás
- Komplex rendszerek értékelésére alkalmas „KRÉTA" programcsomag
- Keresetszabályozás
- AMÉTA export elszámolás
- Létesítményjegyzék szerkesztő program
- Tápkeverő program
- Rendelés és készletnyilvántartás
- Statisztikai programcsomag
- Nyílt árusitású üzletek értékesitése és készletnyilvántartása
- Nagyüzemi keresetadó számitás
- Névsorkészítés
- Címletezés
- Bérstatisztikai és címletező rendszer
- Matematikai programcsomagok
- KPM statisztika
- Információs program

TPA 1140/48/440 számítógépekre:

- MTEACP
- Matematikai Statisztika
- INVESZT-A beruházások pénzügyi folyamatának nyilvántartása
- SELECT általános leválogató program
- TABGEN általános táblageneráló program
- MULTIFMS többterminálos formátumkezelő rendszer
- Magyar nyelvü szövegszerkesztő
- Intézményi költségvetési operativ információs rendszer
- AIE-HAHÓ univerzális párbeszédes magyar nyelvü, többterminálos, ad-hoc lekérdező, naprakészre hozó, listázó programcsomag
- MUV lemezfüző szoftver

Számítástechnikai Iroda
Budapest, V. Garibaldi u. 2.
Levélcím: 1519 Budapest, Pf.: 330
Telefon: 112-660, 113-608
Telex: 22-7272

Szluka Emil–Pető Gábor Pál: A számítógépről mindenkinek

(Budapest, 1984. Népszabadság – Hírlapkiadó, 130 oldal. Ára: 22,–Ft.)

A szerzők a számítástechnikai alapoknak, elemi ismereteknek a legszélesebb olvasóközönséggel való megismertetését tűzték ki célul. A feladat nagy felelősséget ró a szerzőkre, lektorokra egyaránt. Hogyan lehet azokhoz közel hozni a számítógépet, akik benne jobbára még mindig misztikus ködbe burkolt, érthetetlen műszaki csodát látnak, de akiknek a mindennapi életébe néhány év múlva feltartóztathatatlanul belép ez a gép, mint ahogy korábban belépett az autó, a repülő, a rádió, a telefon? Hol kell meghúzni a határt a jó szándékú ismeretterjesztés és a szakmai pontosság, részletezés között?

A számítógépről mindenkinek csak a „hasonló módon, mint ha..." vagy „olyan, mint..." tárgyalásmódban lehet és kell írni, amellett, hogy bizonyos alapfogalmak pontos tisztázása elkerülhetetlen. A kritikus nem kérheti számon az ilyen céllal írt műtől a szakkönyvek egzaktságát, de két dolgot feltétlenül meg kell követelnie: a didaktikai felépítés kikezdhetetlenül átgondolt logikáját és azt, hogy az egyszerűsítő szemléletmód ne csapjon át téves értelmezést sugalló vagy egyenesen egyértelműen hibás tanokat tartalmazó fogalmazássá. Csak az vállalkozzék a számítástechnika ilyen jellegű népszerűsítésére, aki tökéletesen uralja a témát.

A könyv szerző és téma szerint két részre oszlik. Az első rész címe: „Hogyan működik a számítógép?" E részben a didaktikai felépítéssel szemben sok észrevételnek nincs helye, mert logikai vezérfonal egyáltalán nem fedezhető fel benne. Csak néhány példa:

– az olvasó előbb értesül a mágneslemez nyomvonaláról és szektoráról (ez utóbbiról minden magyarázat nélkül), mint arról, hogy mi a bit és a bájt;

– akkor, amikor a szerző kijelenti, hogy „előttünk áll tehát, még ha csak nagy vonalakban is, az a gépi együttes...", az olvasó túl van már a lyukszalagos kódon, a tárakon, az olvasó- és beírófejeken, egy hosszú, fáradságos és nehezen érthető értekezésen a fényszedésről, mint a digitalizálás mintapéldájáról, de még nem olvasott arról, hogy a központi egység, a processzor tulajdonképpen mire való;

– amikor a gépi adatfeldolgozás rövid összefoglalását adja a szerző, olyan fontos fogalmakat még nem tisztázott, hogy mi az a gépi utasítás, mi az a számítógépes program;

– érthetetlen, hogy ha már az ismeretlen francia ORIC gépet választotta ki a BASIC nyelvvel való „eljátszadozáshoz", akkor miért nem szerepel ez a típus az előző fejezet („Melyiket válasszam?") összehasonlító táblázatában megadott gépek között?

A példák sorozata folytatható lenne. És nehéz számba venni a szakmai tévedések, a hibás értelmezésnek tápot adó fogalmazási módok, a rosszul vagy egyáltalán nem megmagyarázott fogalmak tömegét.

Furcsa matematikát ismer a szerző. Olyat, amelyben százszor egy tízezred másodperc egy másodpercet tesz ki, és amelyben „meglepi" az embert, hogy 4+3·2 nem 14-et, hanem 10-et ad eredményül (már harmadik általánosban bajban van az a nebuló, aki ezen meglepődik!).

Javasoljuk a szerzőnek: próbálja ki, valóban csak álom-e, hogy bebillentyűzhesse egy személyi számítógépen a PRINT "ÜDVÖZLET" utasítást. Megjegyezzük, hogy ha véletlenül az ékezetes karakterek hiányára gondolt volna, készült Magyarországon olyan gép, amely az összes magyar ékezetes karaktert ismeri.

Ne szóljunk itt a stílus nehézkességéről. De arról igen, hogy az olvasó meglepődhet azon, hogy miért nem lehetett egységes írásmódot alkalmazni (STRING – SZTRING, byte – bájt), és esetleg meghökken az „autput (ejtsd: autput)" kifejezésen is. Visszatérve az ORIC géphez, nem értjük, hogy a BASIC bemutatásához miért kellett francia nyelvű példához nyúlni. De ha már így történt, ügyelni kellett volna a helyes fordításra (mióta jelent a „joie" narancsot?), a francia nyelvtan és helyesírás szabályainak betartására!

A második rész a számítógépek alkalmazási lehetőségeit mutatja be. Nem elméleti fejtegetések, hanem nagyon helyesen, az egyes alkalmazási területekről vett példák segítségével.

A szerző érezhetően elemében van, ezzel magyarázható a kissé eklektikus összhatás. Az olvasó, miközben megismerkedik a müon-projekttel, a perisztaltikus pumpával, az epikrízissel és egyéb fogalmakkal, azt a benyomást nyeri, hogy a számítástechnika modern korunkban a bölcsek köve. Nem kell hozzá más, mint hogy a „legjobb, legnagyobb, legkorszerűbb" számítógépekre megírjuk a programokat, és a tudósok, a szakemberek, a vezetők, az orvosok, a közvetlen termelésben résztvevők mind-mind egyszeriben megkapják azt, amire várnak: a kvarkok előírt pályán repülnek, a gyárakban a termelés szervezettsége nő, az állatok jóizűen fogyasztják optimálisan adagolt takarmányukat, a betegek meggyógyulnak – vagy legalábbis nem halnak meg.

A szerző csak akkor árulja el, hogy teljesen tájékozatlan a témában, amikor ilyen mondatokat ir le: „A gép memóriája egy megabyte, ami virtuális tárakkal 4 Mb-ra bővíthető: a tárak mágneslemezek és mágnesszalagok." (Sic!)

Erről a könyvről legyen elég ennyi. A szerzők egy helyen azt a megállapítást teszik, hogy „az információ érték, s mint ilyen, áru is". Az olvasó a könyv kapcsán azt a tanulságot vonhatja le, hogy a rossz információ, ha értéknek nem is érték, de árunak lehet áru.

VÁRGEDŐ TAMÁS

Az adatfeldolgozás fogalommeghatározásai és többnyelvű szótára. MSZ Szabványgyűjtemények 75. (Szerk. Justin Antal Lekt. Győri János)

(Budapest, 1984. Szabványkiadó, 752 oldal. Ára: 430,– Ft.)

A számítástechnikai kultúra széles körű hazai elterjedésének egyik fontos tényezője az egységes, magyar nyelvű szókincs kialakítása és alkalmazása. Ennek elősegítésére a Magyar Szabványügyi Hivatal 1978 óta átfogóan egységesíti a számítástechnikai fogalmakat.

Ez a könyv az eddig megjelent 28 szabványkiadvány anyagát foglalja egységes szerkezetbe. Három részből áll: 1. magyar nyelvű értelmező szótár, 2. ötnyelvű (angol–francia–magyar–német–orosz) összehasonlító táblázatok, 3. betűrendes névmutatók.

Az értelmező szótár és az összehasonlító táblázatok a következő szakterületi csoportosításban közlik az anyagot: alapfogalmak; aritmetikai és logikai műveletek; műveletechnika; adatszervezés; adatábrázolás; adatelőkészítés és adatkezelés; digitális számítógépek programozása; feldolgozási technika; vezérlő, be/kiviteli és aritmetikai eszközök; adathordozók, tárak és kapcsolódó berendezések; megbízhatóság, karbantartás, készenlét; információelmélet; analóg számítástechnika.

A szótárrész az MSZ 7788 „Az adatfeldolgozás fogalmai" szabványsorozat eddig megjelent szabványainak anyagát tartalmazza. Összesen mintegy 1100 fogalom elnevezése és értelmezése (meghatározása, definíciója) található e részben. A fogalommeghatározások tartalmilag megegyeznek az ISO Nemzetközi Szabványügyi Szervezet vonatkozó előírásaival, és ez megkönnyíti a nemzetközi együttműködést ezen a fontos területen. Minden szótári fogalomnévhez hatszámjegyű nemzetközi azonosító tartozik, amely lehetővé teszi a szakkifejezések egyértelmű meghatározását a különböző nyelveken. A szabványsorozat kidolgozásában számos vállalat, intézmény számítástechnikai szakembere vett részt. Az anyagot nyelvhelyességi szempontból az MTA Nyelvtudományi Intézete véleményezte. E sokoldalú egyeztetés eredményeként a szakmai és a nyelvi szempontokat egyaránt figyelembe lehetett venni.

Az összehasonlító táblázatok az MI 7798 „Az adatfeldolgozás többnyelvű szótára" műszaki irányelvsorozat eddig megjelent anyagát foglalják össze. Az értelmező szótár résszel azonos felépítésben közlik a fogalmak nemzetközi azonosítóját és megnevezését öt nyelven, azonban a fogalom értelmezése (definíciója) nélkül. A táblázatok angol és francia nyelvű része az ISO nemzetközi szabványok elnevezéseivel azonos, a német és az orosz nyelvű rész pedig KGST-szabványok, nemzeti szabványok és egyéb forrásmunkák felhasználásával készült.

A betűrendes mutatók nyelvenként tartalmazzák a fogalmak összesített jegyzékét.

A kiadvány hasznos segédeszköze lehet a számítástechnikusoknak, fordítóknak, szakmai szerzőknek, mindenkinek, aki kapcsolatban áll e szakterülettel.

Beszerezhető a Szabványboltban: Bp. VIII., Üllői út 24.

Végül felhívjuk a figyelmet arra, hogy a szabványosításról szóló miniszertanácsi rendelet értelmében a szótárrészben (illetve az MSZ 7788 szabványsorozatban) szereplő fogalmakat csak a közölt (a szabványban meghatározott) értelemben szabad használni.

GYŐRI JÁNOS
Magyar Szabványügyi Hivatal

Etüdök személyi számítógépekre Szerkesztette: Votisky Zsuzsa

(Budapest, 1984. Gondolat Kiadó, 166 oldal. Ára: 67,– Ft.)

Egy új ipari forradalom küszöbén állunk – tartják a szakemberek. Már nemcsak erőnket felülmúló, hanem bizonyos értelemben szellemünket is meghaladó gépek állnak rendelkezésünkre.

Az olcsó személyi számítógépek megjelenésével egyre többen tudnak megfelelni a számítógépek kihívásának, kerülhetnek gépközelbe. Az ismerkedés módja „személyessé" válhat, a gépre vitt problémák nem kapcsolódnak sem mérnöki-tudományos, sem gazdasági feladatokhoz.

A könyv ilyen céllal készült: szerzői azt szeretnék megmutatni az Olvasónak, mi mindenre használható egy személyi számítógép. Többek között grafikára, oktatásra, játékra, mindennapi unalmas rutinfeladatok megoldására. Ilyen témájú programokat és programkészítési ötleteket közölnek, szinte csak a középiskolás matematikára támaszkodva, és az Olvasó felfedező kedvére számítva. Kitekintést adnak a BASIC-től különböző személyi számítógépes nyelvekre: a LOGO-ra és a FORTH-ra. Sok ábrával és rajzzal illusztrálják a programok gondolati vázát és a képernyőn megjelenő végeredményt. A kötet végén rövid összefoglaló áll a BASIC nyelvről és „nyelvjárásairól". (Ismertetés a könyvből)

Speciális igény speciális üzlet **ÁPISZ** Számítástechnikai szaküzlet

Bp. XI., Budafoki út 7. Telefon 665-503

A Számítástechnika Alkalmazási Vállalattal közösen üzemeltetett

boltban

- számítástechnikai médiák, tartozékok
- mágneses háttértárak (mágnesszalagok, lemezek, floppyk 150,– Ft-tól)
- festékszalagok
- ZX Spectrum 16–48 k gépekhez játékprogramok
- mikroszámítástechnikai könyvek
- számítástechnikához kapcsolódó speciális íróeszközök, fóliára író filciron, vonalzók, sablonok stb.
- mikroszámítógép
- szoftvertermékek (operációs rendszerek, programok)
- számítástechnikai szolgáltatások.

Kellemes környezetben hétfőn és szerdán kooperációs konzultációra kedden és csütörtökön szakmai tanácsadásra biztosítunk lehetőséget, délutánonként 14 és 17 óra között. Mind a nagyközönség, mind a vállalatok, intézmények rendelkezésére állunk. Nyitvatartás: hétfőtől – péntekig 9 órától 17 óráig.

Körültekintő kiválasztással tartós használat esetén is biztosítják a sikert az ITV által forgalmazott import mikrográfiai termékek

- könnyen hozzáférhető, cserélhető alkatrészek
- 16 mm-es film, mikrokártya és mikrofiche egyaránt használható
- jó olvashatóság, állítható képélesség
- különböző nagyítást biztosító optikák
- tökéletes megbízhatóság

Garanciális időn belül
és azon túl is
a szerviztevékenységet ellátja az

INFORMÁCIÓTECHNIKAI VÁLLALAT
Kereskedelmi Főosztály
Budapest III., Kerék u. 6.
T.: 803-294

Szelet-fejlődés

A mikroelektronikai gyártás fejlődését a félvezető szeletek átmérőjének növekedése is jellemzi. A hetvenes évek közepe óta a legtöbb gyártó üzemben 75 mm-es szeleteket gyártottak, majd ugyanazon gyártóberendezések felhasználásával fokozatosan áttértek a 100 mm-es szeletekre. A 125 mm átmérőjű szeletek gyártásához azonban már teljesen új gyártóberendezés szükséges. Nyilvánvalóan ez az oka annak, hogy világviszonylatban mindössze kb. 20 vállalat állít elő 125 mm-es szilíciumszeleteket.

A japán gyártó vállalatok a 125 mm-es szeletátmérőt kívánják szabványosítani, az USA várhatóan ezt a fokozatot átugorja, és a 150 mm-es szeletek technológiájára koncentrálja erőit. 1983 vége óta már működik egy 150 mm-es szeleteket gyártó termelősor. A 200 mm-es szeletátmérőre előreláthatólag 1986-ban térnek át.

Zeneszintetizátor

Az elektronikus orgonáiról világhírű nyugatnémet Versi cég Music 64 néven zeneszintetizátort hozott forgalomba a Commodore 64 géphez. A kb. 10 000 forintnak megfelelő összegért kapható billentyűzet és szoftver legnagyobb hátránya, hogy egyszerre maximum csak 3 hangot tud megszólaltatni. Érdekessége viszont, hogy 13-féle hangszer, így például a gitár, a hegedű, a klarinét és a harmonika hangját is képes szintetizálni.

Új irány?

Bár 1984-ben még csak csírájában jelentkezett hazánkban az új irányvonal, az Apple II kompatibilis gépek vonala, mégis magában hordozza a kibontakozás lehetőségét, tovább tagolva az egyébként sem túl homogén hazai piacot.

A szocialista országokban ezt az irányt 1982-ben Bulgáriában indították el az IMKO–2 és a Pravec–82 gyártásával, és az utóbbit 1984-ben továbbfejlesztették Pravec 8B és Pravec 8M néven. Az első két típust elsősorban iskolagépnek szánták, néhány ezret gyártottak belőlük. A második kettőt már professzionális gépként hirdették meg.

E bolgár kezdeményezéseknél sokkal jelentősebb az a szovjet döntés, melynek nyomán AGAT néven Apple II kompatibilis háziszámítógépek gyártását indították be, szovjet mikroelektronikai bázison.

A hazai AX–II nevű, Apple II kompatibilis gépet szintén 1984-ben jelentették be, az Alkotó Ifjúság Egyesülés menedzselésében. A VIDEOTON Rt. 1984. novemberi bejelentése szerint pedig 1985 első felében 100 darab IVEL Z–3 típusú Apple IIe kompatibilis mikroszámítógépet szándékozik a hazai piacon értékesíteni.

Sakkszámítógép

Az Erfurti Mikroelektronikai Kombinátban (NDK) készült el a szocialista országok első sakkszámítógépe. Az idén várhatóan a hazai boltokban is megjelenik kb. 6000 forintos áron.

Új generáció a játékprogramokban

Alig több, mint két éve, 1982 novemberében jelent meg a színes lehetőségekkel is rendelkező házi számítógépek két „klasszikusa", a Sinclair Spectrum és a Commodore 64. Bár ezek a gépek igen sokoldalúak, a családi méretű nyilvántartási és pénzügyi feladatok ellátása mellett leginkább a legifjabb generáció használja őket, elsősorban játékra. Így az is érthető, hogy különösen a játékprogramoknál észlelhető rendkívüli fejlődés.

E géptípusok életének első éve a próbálkozások jegyében telt el, akkor tanulgattuk a gépek biztosította lehetőségek alkalmazását, kihasználását. Az ekkor forgalomba került játékprogramokban még alig van egy-két szinfolt, a hangeffektusok szolidak, a játékok könnyen áttekinthetők, a képernyőn kevés, inkább csak az ember által kezdeményezett mozgás van. Példaképpen említhetjük erre a hazánkban is elterjedt Jetpac, Meteorit, Bomber, Raiders programot.

1983 második fele igazi áttörést jelentett. Az új játékok tobzódnak a színekben, rendkívül mozgalmasak, a hanghatások méltó kíséről a kiteljesedett képernyőnek. A nálunk elterjedtekből ide sorolhatjuk a Scuba, a Flag, a Jetman programot, és nem kis büszkeséggel a hazai sikerszoftvereket, például a Ceasar, a macska nevűt is. Érdekesség, hogy ez utóbbi is, mint a sikerprogramok többsége, mind a Spectrum, mind a Commodore 64 gépen futó változatban elkészült.

Gyógyító játékok

A kaliforniai Stanford Egyetem orvoskutatói megállapították, hogy a számítógépes játékok nemcsak szórakozásra alkalmasak, hanem a gyógyítás elősegítésére is. A kórházból hazaengedett cukorbeteg gyerekek körében alkalmazták a mikroszámítógépeket, amelyek programjai játékos formában megismertették a gyerekeket betegségük lényegével, problémáival, segítették őket beilleszkedni az életbe. A kidolgozott játék főképpen a 9–14 éveseket vonzza, és felkészíti őket arra, hogyan kell élniük krónikus betegségükkel.

A számítógépes játékok azonban másra is jók. Egy bristoli (Anglia) neurológiai intézetben arra használják őket, hogy segítsenek meggyógyítani a közlekedési balesetekből vagy sportolás közben szerzett sérülésekből visszamaradó fejfájásokat. Az ilyen betegekkel sokszor ugyanis az a gond, hogy a kórházból hazatérve, nehezen tudják figyelmüket összpontosítani, ezért hajlamosak arra, hogy szinte egész nap a tévé előtt üljenek. Ha már ott ülnek – mondják az orvosok –, akkor legyenek aktívak, és játsszanak.

Napi háromórás ilyen játékprogramok segítségével elérték, hogy a betegek mozgásában érezhető javulás mutatkozott: a mozgás összehangoltsága, a végtag- és a szemmozgások, az emlékezet jobb volt, mint azoknál a betegeknél, akik nem kaptak ilyen utókezelést.

A bristoli kísérlet eredménye biztató, és valószínűleg előbb-utóbb általánosan bevezetik utókezelési célra ezt a módszert.

Mikrogépek kooperációban

Már az első félévben mintegy száz darab Z–3 típusú mikroszámítógépet kíván értékesíteni a Videoton Rt. a hazai piacon. Ez a gép a jugoszláviai IVASIM céggel kooperációban készül, a jellegzetes formatervezésű Videoton terminálokba építik be Jugoszláviában. A gyártás mennyisége hazai szemmel nézve nagy, már tavaly is száz készült a gépből havonta.

A Z–3 az Apple IIe géppel kompatibilis. Egy Z80 és két Rockwell 6502 típusú mikroprocesszort tartalmaz. Az Apple DOS 3.2/3.3 verzióján kívül a CP/M és az USCD p rendszere is fut rajta. Csupán tájékoztató jelleggel megemlítjük, hogy a 128 kbájtos központi egység két hajlékonylemezes tárolóval és színes illesztővel Jugoszláviában 1,9 millió dinárba (kb. félmillió forint) kerül.

Értesítjük Tisztelt Vásárlóinkat, hogy a Flórián térnél:

BUDAPEST III., KERÉK u. 4. sz. alatti MINTABOLTUNKBAN

nagy választékban kaphatók:

integrált áramkörök, tranzisztorok, diódák, trafók, relék, csatlakozók, elemek, gyengeáramú akkumulátorok, kazetták, műszerek, jelzőlámpák, biztosítékok, számítógép-alkatrészek, kondenzátorok, mágneses adatrögzítők, személyi számítógépek.

Részletes tájékoztatással, tanácsadással készséggel áll felhasználóink rendelkezésére
Kereskedelmi főosztályunk
Telefon: 803-294
Levélcim: 1369 Budapest, Pf. 314

INFORMÁCIÓTECHNIKA VÁLLALAT
Központ: Budapest V., Bécsi utca 8.
Telefon: 184-899
Telex: 22-4381; 22-6841

Nagy teljesítményű,
szép írásképű,
megbízható nyomtatót kíván használni
számítógépéhez?

Dolgozzon
a
Magyar Optikai Művek

ROMOM k6311

típ. mátrixnyomtatóval

Kapcsolható
a Commodore gépekhez
és minden olyan számítógéphez,
amelynek Centronix illesztője van

A kedvező árral
és szolgáltatási feltételekkel
rendelkező nyomtatót 1985-től
folyamatosan tudjuk szállítani vevőinknek

Gyártja:

Magyar Optikai Müvek
Budapest XII., Csörsz u. 35.
T.: 151-230

MOM BUDAPEST

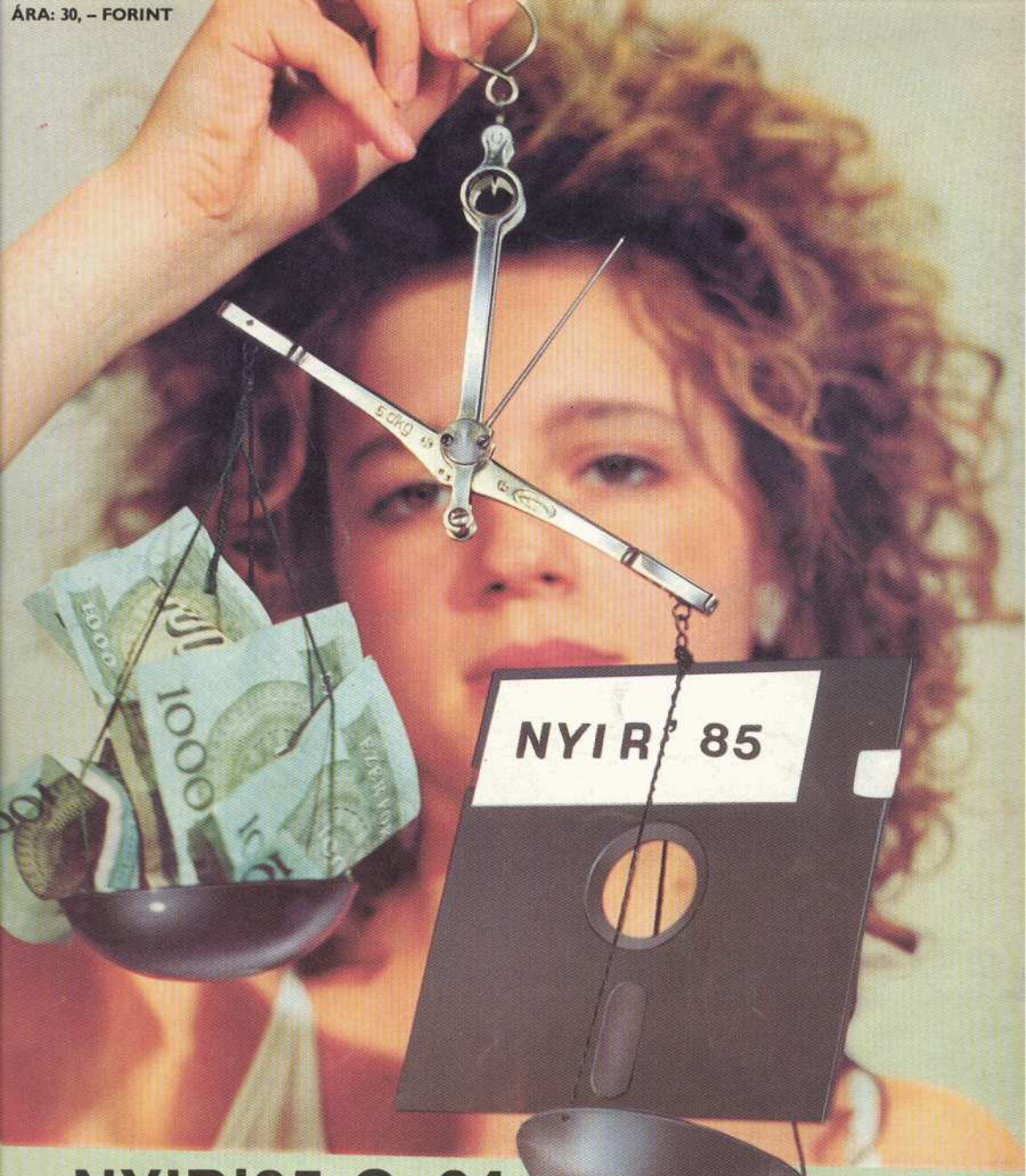
Főbb jellemzői:	
nyomtatási sebesség	**100 jel/s**
nyomtatási raszter	**9 × 7**
írásformák	**ferde írás, normál írás**
nemzetközi karakterkészlet	**szabadon betölthető**
nyomtatási pozíciók soronként	**80 jel/10 cpi**
	100 jel/12,5 cpi
	120 jel/15 cpi
sortávolság	**6 sor/inch**
papírtovábbítás	**súrlódóhengeres tüskés rendszerrel**
másolatok száma	**1 eredeti, 2 másolat**

Forgalmazza:

MIGÉRT

Műszer és
Irodagépértékesitö
Vállalat
Számítástechnikai és
Ügyvitelgépesitési Osztály
Budapest VI.,
Népköztársaság útja 2.
T.: 323-332

ÁRA: 30, – FORINT



NYIR'85 C–64

számítógépre készített 1985. évi vállalati jövedelemszabályozási összefüggéseket tartalmazó interaktív rendszer szolgáltatásai:

– keresetiadó-számítás az összes keresetszabályozási változatban,
– érdekeltségialap-számítás,
– nyereségszámítás és felosztás

KERSZI

AZ ÖNÖK VÁLLALATÁRA ADAPTÁLJUK